Sistema de classificació automàtica per a continguts audiovisuals mitjançant Low-Level Descriptors d'MPEG-7

Projecte Final de Carrera

Estudiants: Borja Gorriz Hernando i Beatriz Martínez Balboa

Tutor: Ignasi Esquerra Llucià

Prefaci

Aquest document és la memòria del projecte final de carrera de la titulació d'Enginyeria Tècnica de Telecomunicacions especialitat a So i Imatge d'en Borja Gorriz Hernando i la Beatriz Martínez Balboa. El projecte ha estat desenvolupat parcialment a Vídeo Stream Networks (VSN) com a conseqüència d'un conveni universitat-empresa, amb la supervisió i ajuda d'Octavi Estapé, tutor a l'empresa del projecte, coordinadament amb el director de I+D, Jordi Gilabert i als laboratoris del Departament de Teoria del Senyal i Comunicacions (TSC) de Terrassa, dirigit per Ignasi Esquerra i Llucià.

La idea del projecte sorgeix de la necessitat de VSN d'obrir noves línies de desenvolupament de software relacionades amb el so i la generació automàtica de dades de descripció del contingut audiovisual que emmagatzemen: les meta-dades Sumat al nostre interès personal en la matèria i la necessitat de realitzar un projecte final de carrera per tal de finalitzar els nostres estudis, sorgeix la idea del projecte que exposem en aquest document.

"Dream in an pragmatic way."

Aldous Huxley

"The truth will set you free. Either that or it'll get you a punch in the nose."

Nick Hornby, A long way down (2005)

"La próxima vez que levantes las cejas de incredulidad, que sea al mundo y no a mí."

Nueva Vulcano, Te debo un Baile

Index

1 Introducció....1

1.1 Objectius....3

1.2 Pla de treball....4

1.3 Estat de l'art....6

1.4 Visió General del projecte....8

2 Estàndard Mpeg-7....9

2.1 MPEG-7 Description Definition Language (DDL)....11

2.1.1 Components estructurals d'XML....12

2.1.2 Tipus de dades en l'esquema XML....13

2.1.3 Extensions d'Mpeg-7....14

2.2 Mpeg-7 Multimedia Description Schemes....15

2.2.1 Organització general de les eines desl Ds....15

2.2.2 Audio Description Schemes....16

2.3 Audio Low-Level Descriptors....19

2.3.1 Paràmetres bàsics i notacions....20

2.3.2 Audio Segment Description i ScalableSeries Description....21

2.3.3 Basic Descriptors....22

2.3.3.1 Audio Waveform....22

2.3.3.2 Audio Power....23

2.3.4 Basic Espectral descriptors....23

2.3.4.1 Audio Spectrum Envelope....23

2.3.4.2 Audio Spectrum Centroid....24

2.3.4.3 Audio Spectrum Spread....24

2.3.4.4 Audio Spectrum Flatness....24

2.3.5 Basic Parameters....25

2.3.5.1 Audio Harmonicity....25

2.3.5.2 Audio Fundamental Frequency....26

2.3.6 Timbral Descriptors....26

2.3.6.1 Temporal Timbral Descriptors....26

2.3.6.2 Log Attack Time....27

2.3.6.3 Temporal Centroid....27

2.3.7 Spectral Timbral Descriptors....28

2.3.7.1 Harmonic Spectral Centroid....28

2.3.7.2 Harmonic Spectral Deviation....28
2.3.7.3 Harmonic Spectral Spread....28
2.3.7.4 Harmonic Spectral Variation....29
2.3.7.5 Spectral Centroid....29
2.4 Silence Segments....29

3 Eines Utilitzades....30
3.1 Entorn de programació NetBeans i llenguatge JAVA SE....30
3.2 FFMPEG....30
3.3 Wavesurfer....31
3.4 Mpeg-7 Audio Encoder....32
3.5 XMLBeans....35
3.6 Data mining: Weka....35

4 Base de dades....43
4.1 Procés de segmentació manual i descripció de la base de dades....44
4.1.1 Segmentació manual i criteris d'etiquetat....45
4.1.2 TestMpeg-7....46
4.2 Estadístiques....48
4.2.1 "A banda i banda"....48
4.2.2 "Set dies"....50
4.2.3 "Naturalment"....50

5 Sistema de classificació....53
5.1 Extracció i adequació de les característiques de les dades....54
5.2 Entrenament i creació dels models....58
5.2.1 Discriminació d'events sonors....62
5.2.1.1 Discriminació silenci-no silenci....62
5.2.1.2 Discriminació veu-música....64
5.2.2 Discriminació del gènere del locutor....74
5.3 Implementació del software....82
5.3.1 Extracció de les característiques i classificació: ExtractDesciptors i MixDescriptors....83
5.3.2 Representació de la classificació i segmentació: ViewDescriptors i MixDescriptors....86
5.4 Esquema general dels sistema....89

6 Experiments i resultats....90
6.1 Mètode d'avaluació de la classificació....91
6.2 Mètode d'avaluació de la segmentació....91
6.3 Avaluació dels classificadors....93
6.3.1 Classificador silenci-no silenci....93
6.3.2 Classificador veu-música....94
6.3.3 Classificador home-dona....96
6.4 Avaluació de la segmentació general....97
7 Conclusions....99
7.1 Futures línies de treball....100
7.2 Opinió personal....100

8 Referències....101

1. Introducció

Amb l'aparició i l'evolució de les tecnologies de la informació, hi ha tals quantitats de material audiovisual disponible que fa impossible trobar tot allò que es necessita. Com a dada, a mitjans de 2007, la coneguda pagina de vídeos Youtube, rebia 6 hores de vídeo per minut, i actualment el material audiovisual que reben els servidors és de 20 hores per minut. Això fa més de 36 milions d'hores de vídeo als servidors. En un altre context, les emissions televisives han augmentat considerablement en els últims anys, i per tant el material a emmagatzemar també.

Aquest augment de suport audiovisual, fa impossible accedir a tot allò que ens interessa al moment que ho volem, i és ben conegut que el valor de la informació depèn de com de fàcil és accedir-hi i gestionar-la. La manera més bàsica d'accedir a la informació és, en el cas d'una cançó a partir del nom d'aquesta i el nom de l'artista. En el cas d'un programa de televisió amb la data d'emissió, el nom del canal,etc. Si parlem de vídeos o arxius sonors qualsevol, la manera d'identificar les dades pot ser molt canviant.

Totes aquestes descripcions resideixen de manera explícita, és a dir, algú ha catalogat les dades i en conseqüència, aquesta creació és un procés subjectiu i vulnerable a interpretacions errònies. En canvi la informació que resideix de manera implícita a les dades es refereix a allò que realment hi ha degut a les característiques de la informació: color, textura, harmonia, etc. A totes aquestes descripcions, les anomenem meta-dades [1][2].

Les meta-dades[3] informen de característiques que resideixen de manera implícita o explícita a les dades i tenen la finalitat de gestionar més eficientment tots els continguts que es generen amb el pas del temps i que, cada vegada en són més. D'aquesta manera s'intenten facilitar i agilitzar tasques de recerca. Tradicionalment, aquestes meta-dades s'han generat de manera manual, sent aquest un procés tediós, temporalment i econòmicament costós, ja que és necessària la presència física d'una persona encarregada. El procés de generar-les automàticament és un altre pas per arribar a transformar les màquines en éssers intel·ligents. És importat l'estandardització i normalització de la manera en què es creen per garantir compatibilitat en diferents sistemes. Existeixen diferents estàndards que tracten el tema. ID3 és un dels més coneguts, permet l'anotació de etiquetes bàsiques tals com intèrpret o nom de la cançó per arxius mp3. Existeixen altres com DublinCore, RDF, etc. En el cas d'aquest estudi, ens centrarem en l'estàndard de MPEG-7[4][5], ja que tal i com s'explicarà posteriorment, és el que més s'adequa a les nostres necessitats.

MPEG-7 és un estàndard de la Organització Internacional per la Estandardització ISO/IEC i desenvolupat pel grup d'experts MPEG, la primera versió del qual va estar aprovada al 2001. Aquest estàndard neix per cobrir les mancances de descripció de contingut audiovisual i creació automàtica de meta-dades, i les dificultats per tal d'ordenar la informació i trobar-la.

Pel que fa a aquest projecte, s'han estudiat una sèrie d'eines proporcionades per l'estàndard MPEG-7 que permeten l'extracció de característiques que resideixen al so. Un cop extretes aquestes característiques utilitzant únicament eines MPEG-7, i en funció de quin volem que sigui el resultat, hem elaborat una serie de classificadors, que posteriorment hem provat i modificat fins a arribar al classificador final.

1. 1. Objectius

Davant de la problemàtica presentada a l'apartat anterior, es defineixen uns objectius en funció del que VSN espera de nosaltres. Pel que fa als requeriments de l'empresa en podem parlar de l'objectiu general i final el qual serà construir un sistema de classificació i detecció automàtic de diferents esdeveniments sonors, a determinar un cop l'estudi hagi avançat i en funció del material audiovisual del qual l'empresa disposa, amb la màxima efectivitat possible. Tot i que els esdeveniments sonors a classificar depenen del material audiovisual del qual disposarem, a priori es poden definir unes fites:

- Construir un discriminador entre veu, música, silenci, i soroll.
- Construir un discriminador de gènere
- Construir un discriminador de diferents locutors
- Construcció d'un segmentador automàtic dels diferents esdeveniments sonors.

Per una altra part, l'objectiu acadèmic i de recerca serà estudiar si es pot construir un classificador eficient mitjançant únicament les eines que proporciona l'estàndard MPEG-7.

1. 2. Pla de treball

El pla de treball que es presenta a continuació correspon al treball desenvolupat per dos enginyers, d'aquesta manera, hi ha part del treball que s'ha repartit i parts que s'han fet independentment.

La primera part de qualsevol projecte contempla la recerca d'informació sobre el tema i de l'estat de l'art de la tecnologia en aquest cas. En el cas particular del nostre projecte, el primer pas serà estudiar l'estàndard MPEG-7, on es pretén assolir els coneixements necessaris per poder desenvolupar tot el projecte. Paral·lelament a aquest procés, ens situem a l'estudi de l'estat de l'art, és a dir, informar-nos de quines són les eines que s'utilitzen actualment, estudis i publicacions on es parli de la problemàtica i software comercial existent. Aquesta part de la recerca ens proporcionarà les eines necessàries per al desenvolupament del classificador a més d'informació de cóm encarar els diferents passos

Un cop finalitzada la primera etapa de recerca, ens endinsem en la creació de la base de dades. La base de dades, ens la proporcionarà l'empresa mitjançant els seus servidors, per tant, una primera part de la creació de la base de dades serà la obtenció del material audiovisual. Un cop tinguem els primers materials, comença un procés llarg i tediós, que és la segmentació manual, on s'hauran de visionar tots els vídeos obtinguts a l'etapa anterior.

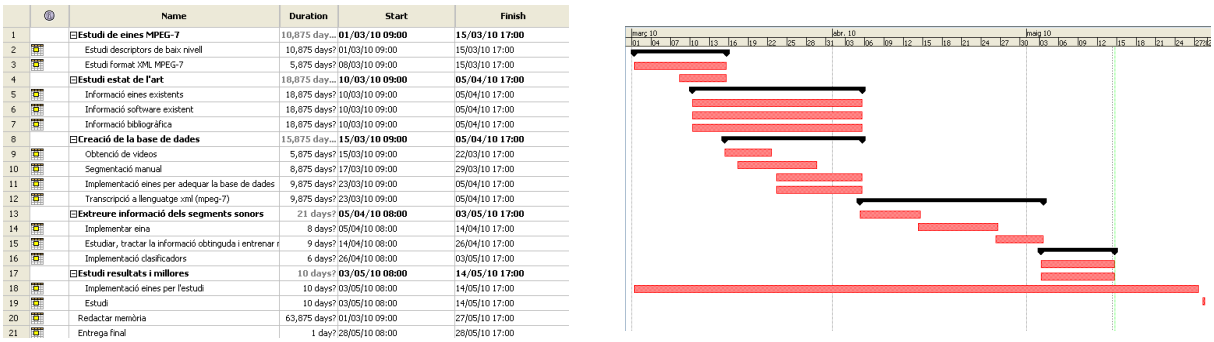
Per altra banda i de manera paral·lela o no, es començarà a desenvolupar els programes que s'utilitzaran per tal d'adequar la base de dades als requeriments dels següents passos A més, el desenvolupament d'un programa per tal de presentar les dades en format XML d'MPEG-7. Amb això finalitza la etapa de creació de base de dades.

La tercera fase del projecte, correspon a l'extracció de la informació que resideix dins dels propis arxius sonors, amb el corresponent tractament de les dades i adequació a altres passos posteriors dels resultats. Un cop finalitzada aquesta part, es pot començar a implementar el classificador.

Pel que fa al desenvolupament de l'algorisme general de classificació, l'últim pas serà provar els classificadors i implementar millores, per tal d'aconseguir un classificador el més eficient possible.

En quant al projecte de recerca, cal un últim pas de presentació de resultats, i creació de últims programes per tal de representar l'efectivitat de l'algorisme, tot això presentat a la memòria que s'ha anat elaborant des del començament del projecte.

Per presentar tot això d'una manera més gràfica, a continuació es presenta el diagrama de Gant realitzat amb el número de dies corresponents a cadascun dels processos.



	Name	Duration	Start	Finish
1	**Estudi de eines MPEG-7**	10,875 day...	**01/03/10 09:00**	**15/03/10 17:00**
2	Estudi descriptors de baix nivell	10,875 days?	01/03/10 09:00	15/03/10 17:00
3	Estudi format XML MPEG-7	5,875 days?	08/03/10 09:00	15/03/10 17:00
4	**Estudi estat de l'art**	18,875 day...	**10/03/10 09:00**	**05/04/10 17:00**
5	Informació eines existents	18,875 days?	10/03/10 09:00	05/04/10 17:00
6	Informació software existent	18,875 days?	10/03/10 09:00	05/04/10 17:00
7	Informació bibliogràfica	18,875 days?	10/03/10 09:00	05/04/10 17:00
8	**Creació de la base de dades**	15,875 day...	**15/03/10 09:00**	**05/04/10 17:00**
9	Obtenció de videos	5,875 days?	15/03/10 09:00	22/03/10 17:00
10	Segmentació manual	8,875 days?	17/03/10 09:00	29/03/10 17:00
11	Implementació eines per adequar la base de dades	9,875 days?	23/03/10 09:00	05/04/10 17:00
12	Transcripció a llenguatge xml (mpeg-7)	9,875 days?	23/03/10 09:00	05/04/10 17:00
13	**Extreure informació dels segments sonors**	21 days?	**05/04/10 08:00**	**03/05/10 17:00**
14	Implementar eina	8 days?	05/04/10 08:00	14/04/10 17:00
15	Estudiar, tractar la informació obtinguda i entrenar r	9 days?	14/04/10 08:00	26/04/10 17:00
16	Implementació clasificadors	6 days?	26/04/10 08:00	03/05/10 17:00
17	**Estudi resultats i millores**	10 days?	**03/05/10 08:00**	**14/05/10 17:00**
18	Implementació eines per l'estudi	10 days?	03/05/10 08:00	14/05/10 17:00
19	Estudi	10 days?	03/05/10 08:00	14/05/10 17:00
20	Redactar memòria	63,875 days?	01/03/10 09:00	27/05/10 17:00
21	Entrega final	1 day?	28/05/10 08:00	28/05/10 17:00

Figura 1.1. Diagrama de Gant corresponent al pla de treball

1.3 Estat de l'art

Les línies d'investigació obertes en el tema de la indexació automàtica actualment són moltes ja que s'obté informació de diferents dispositius i la manera de tractar-la, per tant, ha de ser diferent. El nostre projecte està orientat a la indexació de la informació sonora. Tot i això, actualment s'estan realitzant estudis per tal de millorar la manera en què s'extreu la informació en el camp del vídeo, la imatge i la transcripció de documents de text, essent ja de gran presència els avenços en el camp del vídeo, com ara el reconeixement facial [6].

Pel que fa al camp de l'àudio[7], hi ha documents des de 1996 on es mostren estudis i projectes d'investigació per tal de generar classificadors automàtics d'àudio. Una de les primeres i més importants tasques que apareixen, és la necessitat de diferenciar parla de música, Speech/Music Discriminator (SMD). La importància de disposar de mecanismes que realitzin aquesta operació no és una altra que servir com a pas previ d'altres aplicacions com per exemple transcripció automàtica de veu a text (Speech-To-Text), o l'anàlisi de peces musicals. Un dels primers treballs adreçats particularment al problema de l'SMD va ser publicat al 1996 [8] al qual s'extreien característiques a partir del numero de encreuaments per zero (ZCR) combinat amb un sistema de classificació. A [9], s'estudia l'eficiència per aquest cas de 13 descriptors freqüencials i temporals combinat també en un sistema classificador Gaussian Mixture Model (GMM) i reducció de dimensionalitat.

Pel que fa a estudis més contemporanis, la mecànica continua essent la mateixa. A [7] es planteja la problemàtica estudiant quatre característiques de l'àudio: numero de encreuaments per zero(ZCR), Spectral Roll-off, Loudness i Freqüències fonamentals, i combinat-les de manera que s'aconsegueix una alta efectivitat. Altres estudis fets mostren l'efectivitat de la utilització de descriptors definits per l'estàndard MPEG-7 per a desenvolupar aquesta tasca[11]

El mateix camí es segueix en [12], on s'extreuen un total de dinou característiques, deu de les quals estan definides per l'estàndard MPEG-7 per tal distingir entre tres cantants xinesos -Wu, Du i Lin- i per una altre banda per identificar diferents instruments musicals -violí, viola, cello, i baix-. S'utilitzen diferents tècniques de classificació, Nearest Neightbor Rule (NNR), Artificial Neural Networks (ANN), Fuzzy Neural Networks (FNN) i Hidden Markov Models (HMM), combinat amb Independent Component Analysis (ICA) per tal de reduir la dimensionalitat. Un algoritme molt semblant utilitzen a [13] i [14] per a desenvolupar un classificador d'àudio per a continguts radiofònics com a projecte final de carrera a la UPC.

Tots aquests estudis estan en gran part influenciats pels estudis fets pel grup d'investigadors del Communication Systems Group de la Technical University of Berlin, autors del llibre de referència principal sobre MPEG-7 i l'àudio[4] i que encara avui dia segueixen en el procés de disseny d'un classificador. [15].

Dins de tots aquests estudis portats a terme, s'han aconseguit grans fites i existeixen varies eines produïdes des de el camp acadèmic relacionades amb el tema. El cas més clar que trobem en **MUVIS** [16] , un sistema complet d'indexació i cerca de material multimèdia desenvolupat per l'institut de processat de senyal de la Tampere University of Technology , a Finlàndia. La part del sistema dedicada a l'àudio [17] tracta el problema de la classificació des de un punt de vista diferent als vistos fins ara, fent una classificació iterativa i en forma d'arbre.

També ,altres universitats i centres d'investigació han desenvolupat eines. La Universitat Pompeu Fabra ha desenvolupat el CLAM, un projecte de creació de llibreries C++ pel camp de l'àudio on s'han implementat varies llibreries per a extreure descriptors[18], tot i que cap forma part de l'estàndard MPEG-7. Qui si ha desenvolupat un software de programari lliure per a extreure els diferents descriptors MPEG-7 és l'Institute of Communications Engineering de la Aachen University a Alemanya en col·laboració amb l' Università Politecnica delle Marche, a Italia. [19]. Aquest software és el que s'ha utilitzat en aquest projecte per a extreure els descriptors tal i com s'explicarà en capítols posteriors. Seguint el camp del MPEG-7, l'Institute of Informatitzi Systems del grup de recerca austríac de Joanneum, ha implementat l'estàndard MPEG-7 en una llibreria C++ [20] de codi lliure . Amb aquesta llibreria s'aconsegueix crear nous descriptors així com escriure, verificar i llegir arxius XML en MPEG-7.

Des d'un punt de vista comercial, tot i que són molts els estudis que s'han anat desenvolupant al llarg dels anys, i que hi ha molt de mercat per a aquest camp, encara no existeix un software comercial definitiu per a la problemàtica. Són alguns els articles que s'ofereixen, però. Un exemple el trobem en la companya de software Austríaca Sail Labs [21], ofereix un codi per a altres empreses que es pot integrar dins d'un software que permet no només la classificació de contingut amb un arxiu de sortida que segueix el format MPEG-7 (tot i que el sistema per a classificar no utilitza els descriptors definits a l'estàndard), sinó que amb un mòdul de reconeixement de la parla permet generar un text a partir de la veu enregistrada.

1.4 Visió general del projecte

La memòria d'aquest projecte està organitzada de la següent manera:

- El capítol 2 introdueix al lector a l'estàndard MPEG-7, donant una visió general a què és i com s'estructura i centrant-se en els descriptors de baix nivell (LLD).
- El capítol 3 presenta les diferents eines que s'han utilitzat en el desenvolupament del projecte.
- El capítol 4 dona una descripció del contingut de la base de dades així com del procés de generació d'aquesta.
- El capítol 5 conté detalladament tot el procés d'entrenament dels diferents classificadors dissenyats en aquest projecte.
- El capítol 6 exposa els resultats de l'avaluació del sistema classificador entrenat al capítol anterior.
- Finalment, el capítol 7 descriu les conclusions extretes a partir dels resultats així com les futures línies de treball que s'entreveuen.

2. Estàndard MPEG-7

El Moving Picture Experts Group (MPEG) és, segons [22], un grup de l'organització ISO/IEC a càrrec del desenvolupament d'estàndards internacionals per a la compressió, descompressió, processament i representació codificada d'imatges en moviment (vídeo), àudio i combinació d'ambdues. Al 2001, van desenvolupar l'MPEG-7, anomenat també Multimèdia Content Description Interface .

Tal com el seu nom indica, la voluntat va ser crear un estàndard per a la descripció de continguts multimèdia. En concret, s'associa a la descripció dels continguts audiovisuals comprimits pels codificadors MPEG-1 (emmagatzema i descàrrega arxius audiovisuals), MPEG-2 (televisió digital) i MPEG-4 (codifica àudio i vídeo en forma d'objectes), però s'ha dissenyat perquè sigui independent del format del contingut. L'aplicació final d'aquest estàndard és molt amplia, tot i que el camp on més s'ha utilitzat fins ara és el camp de la indexació i retrobament de dades.

Per a dur a terme aquesta feina, es van definir un conjunt de descriptors per a diferents tipus de continguts multimèdia així com el llenguatge i els mètodes per a poder definir descriptors propis, a més de l'estructura de com han de ser i relacionar-se entre sí aquests descriptors.
En concret, els elements principals de l'estàndard segons [4] son:

- **Els descriptors (D):** Donen els valors d'una característica definida, com pot ser la freqüència fonamental de l'àudio.
- **Els esquemes de descripció (DSS):** Especifiquen l'estructura i la semàntica de la relació entre descriptors i també entre altres esquemes de descripció.
- **El llenguatge de definició de la descripció (DDL):** Un llenguatge basat en XML que especifica esquemes de descripció permetent la extensió i modificació dels esquemes de descripció existents.

La relació entre ells es pot veure en la figura següent, extreta de [22]

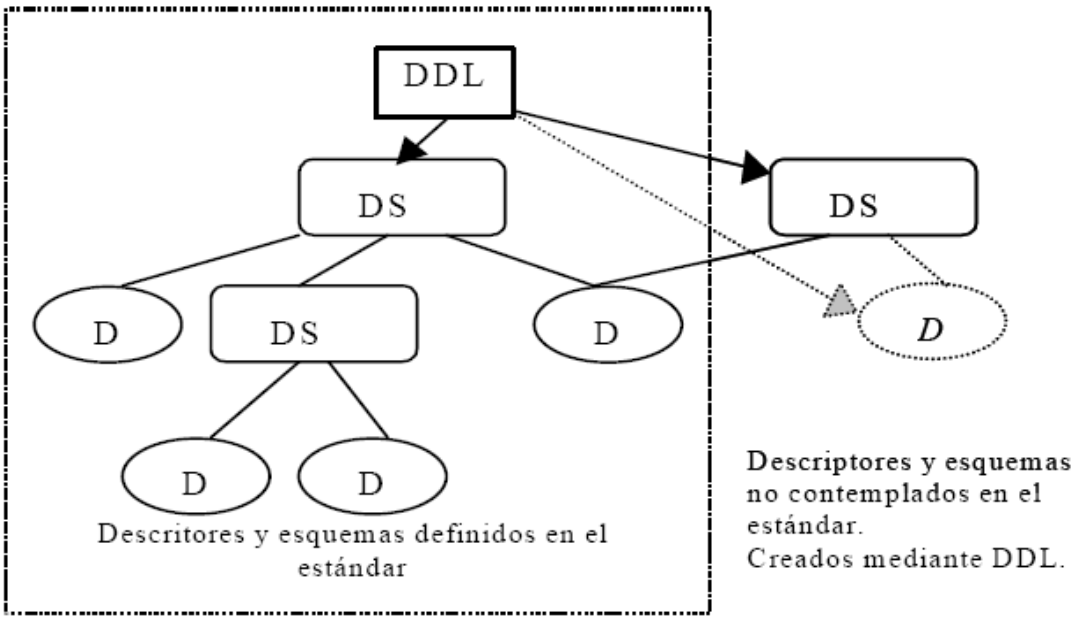


Figura 2.1 Estructura dels diferents elements MPEG-7

Com totes les especificacions MPEG anteriors a la 7, aquesta també té el seu contingut separat per parts. De la llista completa de les parts, per al desenvolupament del projecte es farà èmfasi en les parts que especifiquen els elements exposats anteriorment:

- **Part 2– Description Definition Language:** Especifica el llenguatge per a definir el contingut d'eines de descripció estàndard així com per a definir noves eines.
- **Part 4 – Àudio:** Especifica els descriptors de baix nivell per a la descripció d'àudio.
- **Part 5 – Multimèdia Description Schemes:** Especifica les eines genèriques de descripció, les de més lat nivell.

A continuació es parlarà amb més detall de cadascuna d'aquestes parts, centrant-se en la part 4 que és amb el que es treballà.

2.1 MPEG-7 Description Definition Language (DDL)

Tal com s'ha explicat anteriorment, El DDL o llenguatge de definició és la base per crear tots els elements dins d'MPEG-7. Tal com s'explica al document de requeriments d'MPEG-7 [23] el DDL és " *un llenguatge que permet la creació de nous esquemes de descripció (Ds) i nous descriptors (D). També permet la modificació i millora dels elements ja existents*".

Per tal de poder servir aquest objectiu, durant el procés de creació del DDL es van acordar tres punts bàsics:

- El primer és que fos un llenguatge senzill, que permetes la seva modificació.
- Un segon punt és la cobertura de totes les necessitats i funcions que s'havien determinat per MPEG-7.
- A partir dels dos primers punts, es va determinar que XML havia de ser la forma de sintaxi per al MPEG-7 DDL.

Tal com explica Jordi Vivancos Marti [24], " *XML es pot definir com una eina, independent de maquinari i programari, per transmetre informació i facilitar la cerca i l'intercanvi de tot tipus de dades i documents multilingües*". Dit d'una altra manera, un llenguatge que descriu la forma de definir informació, justament el que els desenvolupadors d'MPEG-7 necessitaven, degut a la seva gran interoperabilitat. Com que MPEG-7 necessita algunes característiques especials, que de per sí XML no incorpora, el DDL es un llenguatge que adopta gran part del esquema XML i l'hi afegeix els mecanismes específics d'MPEG-7.

En concret, MPEG-7 DDL consisteix en tres elements [25]:

1) Components estructurals d'XML.
2) Tipus de dades de l'esquema XML.
3) Extensions d'MPEG-7.

2.1.1 Components estructurals d'XML.

L'esquema XML[25] té definida una forma clara d'estructura les dades, que tothom ha de seguir sigui quina sigui la utilització de la seva sintaxi. Aquest esquema està constituït per tres categories de components: primaris, secundaris i els anomenats "d'ajuda".

Els components primaris són les bases de la sintaxi, l'estructura principal:

1) namespaces i schema wrapper: Inici del document XML. Conté la informació de l'esquema utilitzat, i altres dades generals.
2) Declaració d'elements: Cada element nou dins el document
3) Declaració d'atributs: Els valors que prenen els elements
4) Definició de tipus de dades: simple; complex; derived, anonymous. Aquests són els quatre tipus de dades amb que treballa XML. En MPEG-7 com que tots els elements són definits per l'estàndard, els tipus seran quasi be sempre complexes.

Els components Secundaris poden no trobar-se dins d'un document XML:

1) Definició de grups d'atributs:
2) Definició de grups de model.
3) Definició d'identificadors i restriccions.
4) Declaració d'anotacions: XML permet introduir textos amb anotacions i comentaris,

Els components d'ajuda no poden fer-se servir sols, la seva funció es contribuir a la definició d'altres components:

1) Anotacions
2) Grup de models
3) Particions
4) wildcards.

2.1.2 Tipus de dades de l'esquema XML.

L'esquema XML treballa amb diferents tipus de dades., les quals s'assignen als atributs definits. Hi trobem per una part els tipus ja definits per l'esquema,anomenats primitius, així com alguns tipus derivats d'aquests primers. També es possible, però, definir nous tipus, que es el que fa l'estàndard MPEG-7, ja que no fa servir cap dels predefinits. A la figura 2.2 es pot veure el llista dels tipus ja definits.

Dades primitives	Dades derivades
string	CDATA
boolean	token
float	language
double	IDREFS
decimal	ENTITIES
timeDuration	NMTOKEN, NMTOKENS
recurringDuration	Name, NCName
binary	NOTATION
uriReference	integer, nonPositiveInteger, negativeInteger, nonNegativeInteger, positiveInteger
ID	long, unsignedLong, short, unsignedShort
IDREF	byte, unsignedByte
ENTITY	date, month, year, century;
QName	recurringDate, recurringDay.
	int, unsignedInt
	timeInstant, time, timePeriod

Figura 2.2 Llistat de les totes les dades primitives i derivades definides per XML

La forma de crear dades derivades es a partir de restringir les possibilitats d'una dada primitiva. Un exemple el trobem a continuació, on la variable altura es defineix a partir de fixar uns floats mínims i màxims.

```
<simpleType name="height" base="float">
    <minInclusive value="0.0">
    <maxInclusive value="120.0">
<simpleType>
```

2.1.3 Extensions d'MPEG-7

Tal com ja s'ha dit, era necessari per als desenvolupadors de l'estàndard utilitzar certs components que no figuren en l'esquema XML, i que per tant s'havien d'afegir:

1) Tipus array i Matrix: Les dades de molts descriptors MPEG-7 retornen una serie de valors que s'han de guardar en taules o matrius. Es per això que el grup MPEG va optar per definir aquests tipus.
2) Referències:Es va definir aquest component per proporciona la forma de comprovar el tipus d'un element referenciat.
3) Dades derivades pròpies : A més a més de les dades derivades que ja es trobem definides en l'esquema general d'XML, el DDL d'MPEG-7 incorpora un tipus de dades derivades pròpies. Un exemples serien el codi del país o la regió, o un punters de temps específics: BasicTimePoint i DurationPoint.

2.2 MPEG-7 Multimèdia Description Schemes

Els esquemes de descripció multimèdia MPEG-7 (DS) són estructures de meta-dades per a descriure i anotar continguts audiovisuals. El DS proporciona una manera estandarditzada d'escriure en format XML els valors relacionats amb la descripció i gestió del contingut audiovisual, aquests últims per a facilitar la cerca, indexació i accés dels arxius.

El MPEG-7 Descriptors són dissenyats per a descriure principalment les característiques de l'àudio de baix nivell (descrites més àmpliament en el punt posterior) o característiques visuals com color, la textura, el moviment, energia d'àudio, etcètera, així com atributs del contingut audiovisual en general, com la posició, el temps, la qualitat, etcètera.

La idea és que la majoria de les característiques dels descriptors de baix nivell (LLD) siguin extretes automàticament per l'aplicació que els utilitzi. D'altra banda, el DS és dissenyat principalment per a descriure característiques del contingut audiovisual de més alt nivell, com regions, segments, objectes, esdeveniments i altres meta-dades . Per a aconseguir descripcions més complexes s'integren múltiples descriptors i Ds junts, i es declaren les relacions entre els components de descripció.

A continuació s'explicarà com s'organitzen en general els Ds i les característiques bàsiques dels Ds relacionats amb l'àudio.

2.2.1 Organització general de les eines dels Ds

MPEG-7 Multimèdia Description Scheme (MDS) es pot dividir en general en varies àrees, tal com mostra a figura 2.2 [4]. , de les quals dues àrees són les principals per a aquest projecte i són les que es tractaran amb més detall : **Elements Bàsics** (Basic Elements) i **descripció de continguts** (Content Description)

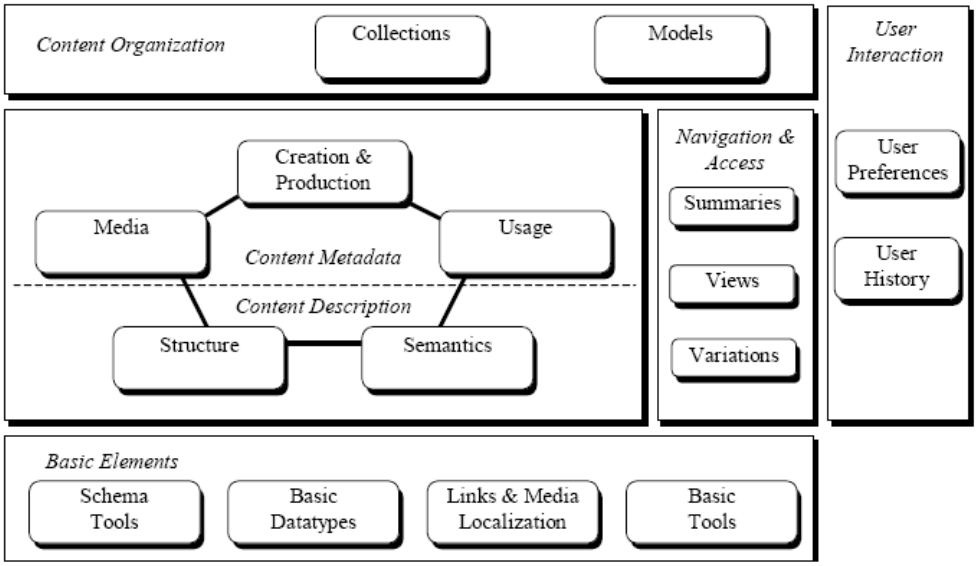


Figura 2.3 Esquema del MPEG-7 Multimèdia Description Scheme, extreta de [25]

·**Elements bàsics :** Esta encarat a les les necessitats específiques de descripció del contingut audiovisual com la descripció de temps, les persones, els llocs i altres anotacions textuals. Com podem veure de la figura 2.2 el conjunt d'elements bàsics estan subdividits en quatre grups: Les eines de l'esquema (schema tools), els tipus de dades bàsiques(Basic Datatypes) , Els localitzadors (Link and Media location) i les eines bàsiques (Basic Tools).

- Les eines de l'esquema s'utilitzen per a crear descripcions valides i gestionar aquestes descripcions. Assisteixen en la formació, empaquetatge i anotació de les descripcions.
- Els tipus de dades bàsiques proporcionen un conjunt de tipus i estructures matemàtiques necessaris per a la descripció. Un exemple serien les matrius i els vectors.
- Els instruments de localització s'utilitzen per a especificar referències dintre de les descripcions, per a connectar les descripcions amb el contingut, per a localitzar el contingut i per a descriure el temps.
- Per últim, les eines bàsiques engloben aspectes de la descripció de continguts multimèdia, com gràfics, anotacions en text, descripció de persones o espai, entre altres.

· **Descripció de contingut:** Aquests Ds Descriuen l'estructura (regions, frames de video i segments d'àudio) i o la semàntica (Objectes, esdeveniments i nocions abstractes).

Per una banda, els estructurals defineixen l'arxiu multimèdia des de la seva estructura. Aquests esquemes utilitzen una descripció a partir dels segments, en el cas de l'àudio. A l'exemple de sota es veu com utilitzarem aquest esquema per a descriure el contingut d'un arxiu d'àudio.

Per l'altra, la descripció semàntica té més a veure amb nocions abstractes del contingut. Aquesta descripció no té interès en el desenvolupament d'aquest projecte.

Figura 2.4 Esquema d'un segment d'àudio segons la descripció de contingut[14]

2.2.2 Audio Description Schemes

La part d'àudio d'MPEG-7 proporciona varis esquemes, construïts a partir de Ds més bàsics, per tal de descriure el contingut d'àudio. Alguns d'aquests esquemes es basen en els descriptors de baix nivell (LLD), que com ja s'ha dit són la base d'aquest projecte i dels quals es parlara més endavant. Altres utilitzen uns altres valors. No obstant aquest conjunt d'esquemes es coneix com descriptors d'alt nivell, que proporcionen descripcions més generals , i els quals ara anomenarem.

Audio Signature Description Scheme

Aquest esquema fa una estadística dels valors obtinguts per el LLD spectral flatness Descriptor , per tal de crear un identificador únic a cada senyal d'àudio. Es tracta de crear una mena d'empremtes dactilars de l'àudio amb el propòsit d'identificar de forma automàtica i robusta els senyals d'àudio. Les aplicacions inclouen la identificació d'àudio basats en una base de dades de treball coneguts.

Eines de descripció del timbre d'instruments musicals

L'objectiu d'aquest descriptor es descriure les característiques perceptuals del so d'instruments musicals utilitzant un nombre de LLD's reduït En el exemple extret de [10] es pot veure una descripció d'un instrument percussiu.

```
<AudioDescriptionScheme xsi:type="PercusiveIntruentTimbreType">
          <LogAttackTime>
               <Scalar> -1,683017 </Scalar>
          </LogAttackTime>
          <SpectralCentroid>
               <Scalar> 1217,341518 </Scalar>
          </SpectralCentroid>
          <TemporalCentroid>

               <Scalar> 0,081574 </Scalar>
          </TemporalCentroid>
<AudioDescriptionScheme >
```

Eines de descripció de melodia

Aquestes eines proporcionen una representació monofònica de la informació de la melodia d'un so per tal de contribuir a una comparació de similituds entre sons eficient i robusta. Aquest Ds inclou un descriptor del contorn de la melodia així com un descriptor de la seqüencial de la melodia per a una representació més acurada.

Eines generals de reconeixement i indexació de so

Són una col·lecció d'eines per a categoritzar el so de forma general. Els Ds proporciona un seguit d'eines automàtiques d'identificació i indexació, a més de l'especificació d'un esquema de classificació de sons i eines per a especificar les classes del sistema. Permet fer una classificació des de descriptors de baix nivell, a descriptors semàntics d'alt nivell passant per models estadístics

Spoken Content Description Tools

El estàndard MPEG-7 proporciona una eina per a la gestió d'aplicacions de reconeixement de la parla, es a dir, extreure informació de veus parlades. La forma en que la descripció es extreta no forma part de l'estàndard. El que proporciona l'eina es una descripció estandarditzada del valors obtinguts per sistema reconeixedor. Com es pot suposar, les aplicacions d'aquesta eina són moltes i molt importants, i és per això que avui dia es treballa en proporcionar sistemes de reconeixement eficients.

2.3 Audio Low-Level Descriptors (LLD)

Els descriptors d'àudio de baix nivell o LLD són d'una importància vital per a descriure l'àudio, ja que com s'ha vist els descriptors d'alt nivell es basen en aquests per tal de extreure la informació. Són per això la base del projecte.

Com es pot veure a la figura 2.5 existeixen un total de 18 descriptors contemplats per l'estàndard.

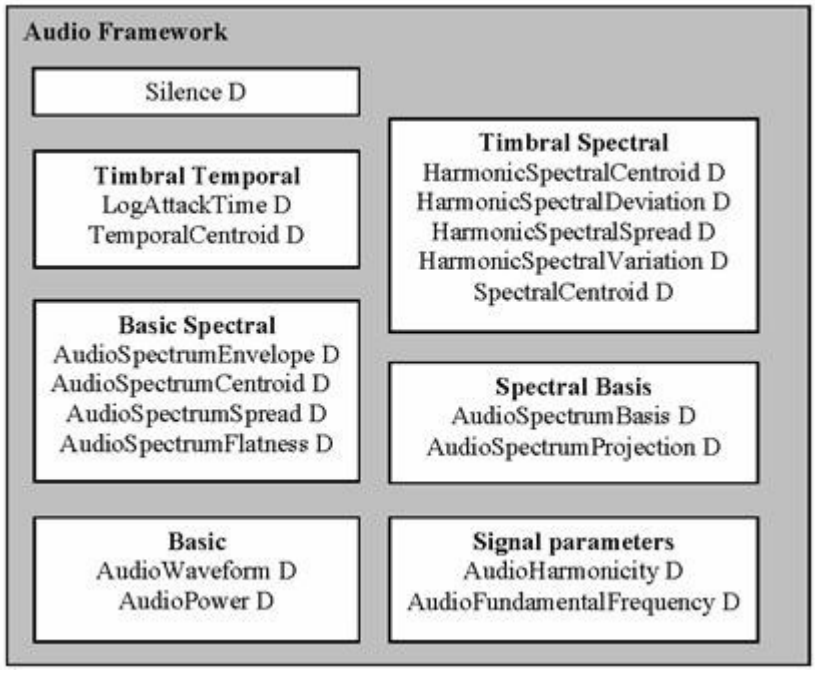


Figura 2.5 Llista de tots els Descriptors de baix nivell d'àudio

Tal com ara veurem, aquests descriptors poden donar-nos informació automàtica de les variacions de l'àudio tant a nivell temporal com freqüencial. Basant-se en aquest informació es com es vol obtenir els models que definiran finalment el sistema classificador.

A continuació es donarà una explicació detallada de cada descriptor, així com una introducció als paràmetres i els termes utilitzats.

2.3.1 Paràmetres bàsics i notacions

A l'hora de parlar de les característiques de l'àudio és comú distingir entre dos dominis, el temporal i el freqüencial. A mesura que es detallin les especificacions de cada descriptors es farà referència a algun d'aquests dominis. Per tal de que sigui fàcil d'entendre, en aquest punt es detallen les notacions utilitzades

Domini temporal:

- *l > Índex de les finestres temporals*
- *hopSize > És l'interval entre dos finestres seguides en el temps*
- *Nhop > Número enter de mostres temporals que corresponen a hopSize*
- *Lw > Tamany de la finestra.*
- *Nw > Mostres corresponents a Lw*
- *L >Numero de mostres totals en s(n)*
- *s(n) > forma d'ona*

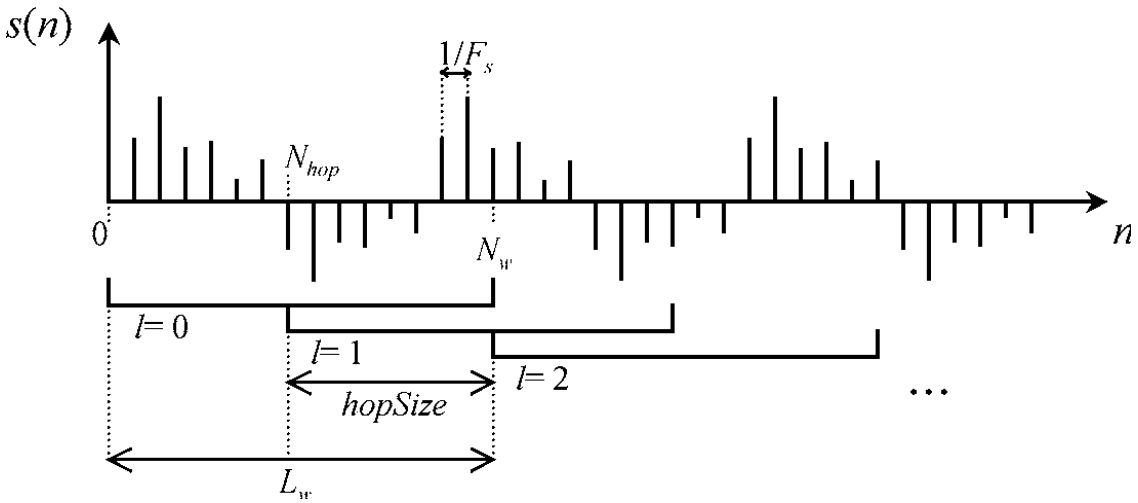


Figura 2.6 Notació per els descriptors que utilitzen finestres

El valor de l'interval i del tamany de la finestra pot variar segons el descriptor que s'utilitzi, a més de poder-se ajustar manualment entre un valors mínims i màxims, tot i que l'estàndard només permet a l'interval ser un múltiple de 10 ms.

Domini freqüencial:

- *k > Índex de les fines tres freqüencials.*
- *Sl (k)> És l'espectre extret del frame l de s(n).*
- *Pl(k) > És el Espectre de potencia extret del frame l de s(n)*

2.3.2 Audio Segment Description i ScalableSeries Description

N'hi ha dues maneres de descriure les característiques de l'àudio segons l'estàndard MPEG-7. Es pot marcar les característiques extretes de sons de longitud variable per a diferenciar regions de característiques diferents. En aquest cas s'està definint una porció del total de l'arxiu, i es guarda com una Audio Segment Description, on el segment pot ser des de una partició arbitraria fins a l'arxiu de so sencer. Aquest model és s'utilitzara en els capítols 4 i 5 per a descriure un arxiu en segments segons la seva classificació i segons l'esquema exposat anteriorment en el punt 2.2.1

També es poden extreure les característiques cada interval de temps per a un fragment de so. En aquest cas, els valors resultants, que anirem obtenint al llarg del temps, es guarden com Scalable Series Descriptions. Aquest tipus de descripcions són les que utilitzarem a l'hora d'extreure característiques de l'àudio a analitzar. Dins d'aquest grup trobem dues variants. Hi ha LlD que retornen un sol valor per a un segment mentre que d'altres retornen una serie. Els primers son els anomenats AudioLLDScalarType, com la freqüència fonamental o l'harmonicitat espectral, mentre que els altres s'anomenen AudioLLDVectorType. Totes dues variants, però, tenen una característica comuna d'escalabilitat. Això significa que es pot variar el nombre de valor que s'agafen per a cada segment, i fins i tot anar variant el valor d'escalabilitat.

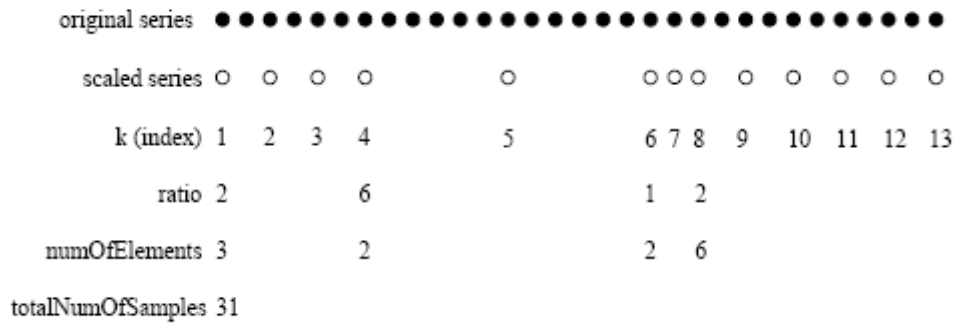


Figura 2.7 Exemple de descripció d'una serie escalada

En aquest exemple veiem com de les 31 mostres originals extretes, la serie escalada extreu un nombre inferior, segons un paràmetre. La primera mostra surt de les dues primeres, però la quarta surt de sis mostres originals. Aquest es el concepte d'escalabilitat.

Podem veure en l'exemple un seguit d'atributs de les series escalades:

- Escala: Indica si com ha sigut escala l'original. En cas de no estar-ho, les mostres originals es descriuen en la seva totalitat.
- Nombre de Mostres : Indica el nombre de mostres originals, abans de l'escalat.
- Ratio: Nombre que indica el nombre de mostres originals representades per una mostra escalada. Si no hi ha escalat, el valor serà 1.
- Nombre d'elements: Indica el nombre final de mostres després de l'escalat. Si aquest fos nul, el nombre d'elements i el nombre de mostres serien iguals.

A llarg del projecte s'avaluaràn i utilitzarant tot tipus de ScalarSeries type, pero en concret els separem en tres:

- Els que retornen un vector, anomenats VectorType.
- Els que retornen un escalar cada cert interval de temps, anomenats SerieOfScalarType.
- I per últim, els que retornen un sol valor per a un segment donat, es a dir que fan el màxim escalat possible.

Aquesta distinció serà important per al bon desenvolupament del projecte.

2.3.3 Basic Descriptors

L'objectiu d'aquests descriptors es proveir una descripció simple de les propietats temporals d'un senyal d'àudio.

2.3.3.1. Audio Waveform

Descriptor que permet obtenir una representació gràfica del senyal en funció del temps. Descriu l'envolvent de la forma d'ona entre diferents segments d'àudio a partir de definir un màxim i un mínim (maxRange i minRange) i dibuixar-los per a cada frame.

```
<complexType name="AudioWaveformType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDScalarType">
                  <attribute name="minRange" type="float"
            use="optional"/>
                  <attribute name="maxRange"type="float"
                  use="optional"/>
            </extension>
      </complexContent>
</complexType>
```

2.3.3.2 Audio Power

Descriu la potencia instantània del senyal. Els coeficients de AP son la mitjana al quadrat dels valors de la forma d'ona s(n) entre successius frames sense overlap (Lw=hopSize).Permet una representació ràpida del senyal.

```
<complexType name="AudioPowerType">
 <complexContent>
 <extension base="MPEG-7:AudioLLDScalarType"/>
 </complexContent>
 </complexType>
```

2.3.4 Basic Espectral descriptors

Quatre descriptors bàsic que proporcionen informació de les series temporal en domini freqüencial logarítmic (aquesta escala s'utilitza per tal de aproximar-se a la resposta de la oïda humana).

2.3.4.1. Audio Spectrum Envelope

S'utilitza per generar un espectrograma reduït de la senyal d'entrada original a partir de l'ús de sub-bandes freqüencials. L'interval es pot escollir entre múltiples d'octava (1/16 d'octava i 8 octaves) o potencies de 2.

```
<complexType name="AudioSpectrumEnvelopeType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDVectorType">
                  <attributeGroup ref="MPEG-
                  7:audioSpectrumAttributeGrp"/>
            </extension>
      </complexContent>
</complexType>
```

2.3.4.2 Audio Spectrum Centroid

Descriu el centre de gravetat de l'espectre freqüencial de la potencia. D'aquesta manera es pot obtenir una visió general de quines són les freqüències dominants en un esdeveniment sonor.

```
<complexType name="AudioSpectrumCentroidType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDScalarType"/>
      </complexContent>
</complexType>
```

2.3.4.3. Audio Spectrum Spread

Aquesta és una altre mesura simple de la forma espectral del senyal. La propagació de l'espectre, també anomenat "ample de banda instantani" (instantaneous bandwith) queda definit per MPEG-7 com el segon moment central de l'espectre freqüencial. Aquest paràmetre proporciona informació referent a si l'espectre està distribuït al voltant del seu "spectrum centroid". Un valor baix significa que l'espectre està concentrat al voltant del centre i, al contrari quan spectrum spread té un valor alt. Aquest paràmetre és útil per tal de diferenciar entre sons amb tons purs i sons sorollosos.

```
<complexType name="AudioSpectrumSpreadType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDScalarType"/>
      </complexContent>
</complexType>
```

2.3.4.4. Audio Spectrum Flatness

Ens diu com és de semblant una senyal d'àudio amb un soroll blanc. Una forma d'espectre plana correspon a soroll o a una senyal impulsiva. Per tant, coeficients ASF alts indiquen sorolls i coeficients amb valors baixos indiquen una estructura harmònica de l'espectre, ja que desviar-se d'una forma plana significa comportament harmònic.

```
<complexType name="AudioSpectrumFlatnessType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDVectorType">
                  <attribute name="loEdge" type="float"
                  default="250"/>
                  <attribute name="hiEdge" type="float"
                  default="16000"/>
            </extension>
      </complexContent>
</complexType>
```

2.3.5 Basic Signal Parameters

Els següents descriptors ofereixen una visió més detallada de l'espectre freqüencial de les senyals d'àudio descrivint el grau d'harmonicitat de les senyals.

2.3.5.1 Audio Harmonicity

Proporciona dues mesures de les propietats harmòniques de l'espectre.:

Harmonic Ratio

És una mesura de la proporció de components harmònics en l'espectre de potència. Ens determina si un so és harmònic o no. (Exemple: comparativa entre: soroll, una flauta, rialles. HR_{flauta}=1, $HR_{rialles}$=1, HR_{soroll}=0,5. (al voltant de).
HR modificat diferencia més clarament el nivell d'harmonicitat .

Upper Limit to Harmonicity

És una estimació de la freqüència a la qual ja no hi ha estructura harmònica. HR juntament amb ULD estan pensades per tal de proporcionar una descripció de les propietats harmòniques del so. Es poden utilitzar per distingir entre sons harmònics (música, veu...), i sons no harmònics (com soroll o segments sense veu).

```
<complexType name="AudioHarmonicityType">
      <complexContent>
            <extension base="MPEG-7:AudioDType">
                  <sequence>
                  <element name="HarmonicRatio" type="MPEG-
                  7:AudioLLDScalarType"/>
                  <element name="UpperLimitOfHarmonicity"
                  type="MPEG-7:AudioLLDScalarType"/>
                  </sequence>
            </extension>

      </complexContent>
</complexType>
```

2.3.5.2 Audio Fundamental Frequency

Proporciona una estimació de la freqüència fonamental f_0 on s'assumeix que el senyal d'àudio es periòdic. És molt útil per tal d'estimar el to (pich) de senyals de música o parla. Existeixen molts algorismes d'estimació de la freqüència fonamental, l'estàndard no especifica un mètode en particular.

```
<complexType name="AudioFundamentalFrequencyType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDScalarType">
                  <attribute name="loLimit" type="float"
            default="25"/>
                  <attribute name="hiLimit" type="float"
                  use="optional"/>
            </extension>
      </complexContent>
</complexType>
```

2.3.6 .Timbral Descriptors

Es coneix com timbre a la característica que permet diferenciar dos sons iguals d'igual to, volum relatiu y duració. L'objectiu d'aquests descriptors es descriure aquesta característica, en principi funcionant conjuntament amb el descriptor de timbre d'alt nivell, tot i que la seva implementació com a descriptors de baix nivell permet el seu ús independent.

2.3.6.1 Temporal timbral descriptors

Proporcionar informació sobre els canvis d'energia de la senyal a partir d'analitzar l'evolució de l'envolvent (la forma) de la senyal. Correspon a descriure les diferents fases d'un so:

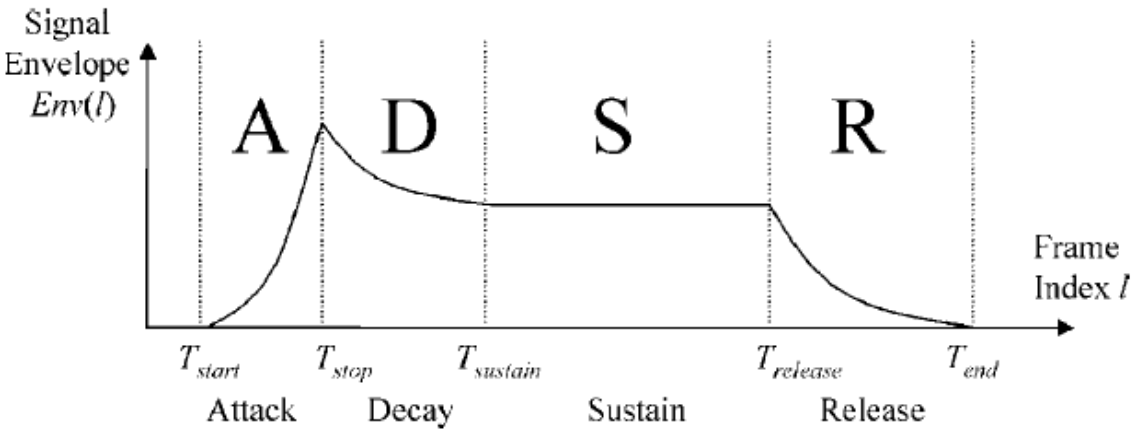


Figura 2.8. Gràfica evolució de l'envolvent d'un so

Attack:(atac) temps que triga el so en arribar al pic màxim inicial de volum.

Decay: (caiguda) temps on la intensitat,després del pic màxim decau fins a un nivell intermig.

Sustain:temps on la intensitat del so es manté constant.

Release:(desaparició)Temps des que la intensitat és intermitja fins que finalment desapareix.

Val a dir que un so no té perquè presentar les 4 fases.

2.3.6.2 Log Attack Time

Proporciona informació sobre el temps d'atac d'un so, a partir de calcular el log attack time (LAT), que es defineix com:

$$LAT=\log(tf - ti)$$

on tf es el moment on el senyal arriba al seu màxim i ti es el temps on el senyal comença (es sol agafar com a valor quan la intensitat supera el 2% respecte a la màxima).

```
<complexType name="LogAttackTimeType">
      <complexContent>
            <extension base="MPEG-7:AudioLLDScalarType"/>
      </complexContent>
</complexType>
```

2.3.6.3 Temporal centroid

Ens dona el valor en temps on trobem el promig energètic de l'envolvent.

```
<complexType name="TemporalCentroidType">
                  <complexContent>
                        <extension base="MPEG-
                          7:AudioLLDScalarType"/>
                  </complexContent>
        </complexType>
```

2.3.7 Spectral timbral descriptors

Proporcionen informació sobre l'espectre harmònic de la senyal, es a dir, sobre el domini freqüencial de la senyal, diferenciant-se dels descriptors freqüencials bàsics (punt 2) en que aquests utilitzen una escala lineal. Requereixen d'un previ anàlisis de la senyal per a poder obtenir el valor i posició dels harmònics freqüencials.

2.3.7.1 Harmonic Spectral Centroid

Dona el valor mig (fent la mitja respecte a la duració de la senyal) de l'amplitud dels pics harmònics que presenta l'espectre. Ens dona informació d'on estan situats principalment els harmònics en l'espectre.

```
<complexType name="HarmonicSpectralCentroidType">
     <complexContent>
          <extension base="MPEG-7:AudioLLDScalarType"/>
     </complexContent>
</complexType>
```

2.3.7.2 Harmonic Spectral Deviation

Proporciona informació sobre el desplaçament dels pic harmònics dins l'espectre.

```
<complexType name="HarmonicSpectralDeviationType">
     <complexContent>
          <extension base="MPEG-7:AudioLLDScalarType"/>

     </complexContent>

</complexType>
```

2.3.7.3 Harmonic Spectral Spread

Ens proporciona informació sobre la propagació de l'espectre respecte a centroide harmònic espectral(punt 4.2.1). Indica si les components espectrals es situen al voltant o no d'aquest punt.

```
<complexType name="HarmonicSpectralSpreadType">
     <complexContent>

          <extension base="MPEG-7:AudioLLDScalarType"/>
     </complexContent>
</complexType>
```

2.3.7.4 Harmonic Spectral Variation

Proporciona informació sobre la variació de l'espectre entre frames adjacents. Permet detectar canvis bruscos en el contingut freqüencial, no lleus moviments, com pot detectar la desviació espectral harmònica (punt 4.2.2).

```
<complexType name="HarmonicSpectralVariationType">
     <complexContent>
                  <extension base="MPEG-7:AudioLLDScalarType"/>
     </complexContent>
</complexType>
```

2.3.7.5. Spectral Centroid

Aquest descriptor no té a veure amb el contingut harmònic de l'espectre. Dona la mitja energètica de l'espectre. Acostuma a donar informació sobre la "brillantor" d'un so.

```
<complexType name="SpectralCentroidType">
     <complexContent>
          tension base="MPEG-7:AudioLLDScalarType"/>
     complexContent>
</complexType>
```

2.4 Silence segments

Aquests descriptor simplement dona una forma definida d'indicar que no es troba cap indici de so en el segment analitzat. El valor confidence reflexa la certesa (de o a 1) amb que es pot afirmar que el segment es veritablement silenci.

```
<complexType name="SilenceType">
     <complexContent>
          <extension base="MPEG-7:AudioDType">
               <attribute name="confidence" type="MPEG-
7:zeroToOneType"                    default="1.0"/>
               <attribute name="minDurationRef" type="anyURI"
                    use="optional"/>
          </extension>
     </complexContent>
</complexType>
```

3. Eines utilitzades

Per tal de desenvolupat tot el projecte hem utilitzat diferents eines ja desenvolupades com a programari lliure en tots els casos. D'aquesta manera, ens evitem repetir feina ja feta per altres. Així doncs, a continuació s'exposa tot el software utilitzat, en primer lloc un entorn de programació i un llenguatge que ens permetrà desenvolupar el projecte, un codificador d'àudio, un programa per tal de segmentar manualment els arxius, una sèrie de llibreries per tal d'extreure informació de l'àudio i tractar-la posteriorment, i per últim una eina de data mining que ens permetrà tractar tota la informació de manera conjunta i elaborar els models.

3.1. Entorn de programació NetBeans i llenguatge JAVA SE

L'entorn de programació on s'han desenvolupat els diferents projectes i programes ha estat NetBeans 6.8[26], programa lliure, amb el llenguatge JAVA SE[27], ja que són les eines que ens van proporcionar VSN.

3.2. FFMPEG

El programa lliure ffmpeg[28] és una eina complerta per processat de àudio i vídeo, on es permet entre altres coses transcodificar els fitxers audiovisuals a partir de les llibreries que incorpora, a més de permetre extreu-re l'àudio de vídeos, retallar segments,etc.

Aquest programa s'executa únicament a partir de línia de comandes, això fa que el seu ús no sigui molt extens. Tot i això, existeixen algunes GUI per ffmpeg i fan molt més interactiva la comunicació amb el programa.

En aquest projecte, ffmpeg s'utilitzarà per dos taques diferents, en primer lloc per extreu-re l'àudio d'un vídeo, i en segon lloc per retallar segments d'àudio. Les comandes necessàries per realitzar aquests processos són les següents:

```
ffmpeg.exe -i video.avi -ac 1 -vn fitxer_sortida.wav

ffmpeg.exe  -i segment_sonor -ss temps_inicial -t    duració+  -vn
fitxer_sortida.wav
```

3.3. Wavesurfer

Wavesurfer [29] que permet visualitzar dades d'àudio i manipular-les. Wavesurfer presenta una interfície simple i de fàcil us, que permet realitzar diferents tasques en àmbits de recerca en el camp de la parla i educació.

En el nostre cas serà la eina que utilitzarem per tal de transcriu-re manualment el contingut de la base de dades que creem. La pantalla d'inici del programa conté una pista d'àudio buida, figura 3.1.

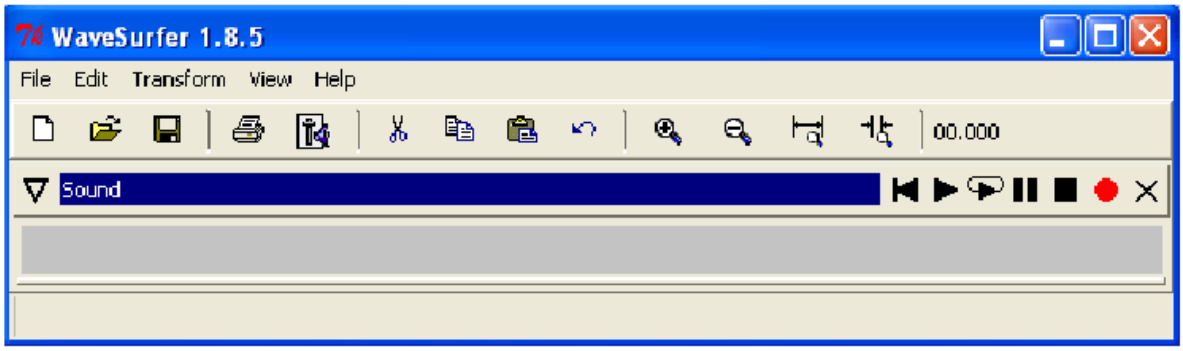


Figura 3.1. Inici programa Wavesurfer

Per tal de realitzar les diferents tasques, el programa permet crear *panes*. Un *pane* és una nova finestra ja a dins del programa que permet seleccionar l'opció de visualització que necessitem, en el nostre cas, serà útil introduir dos o tres finestres diferents, amb per exemple l'espectrograma, la forma d'ona i la transcripció, com es mostra a la figura 3.2.

Pel que fa als formats d'entrada, el programa suporta els formats WAV, AU, MP3, AIFF, CSL i SD. I en quant a la sortida de les dades corresponents a la transcripció, el format és el presentat a la figura 3.3, on apareixen tres columnes corresponents a l'etiqueta donada, a l'instant inicial, i a l'instant final amb l'extensió .lab.

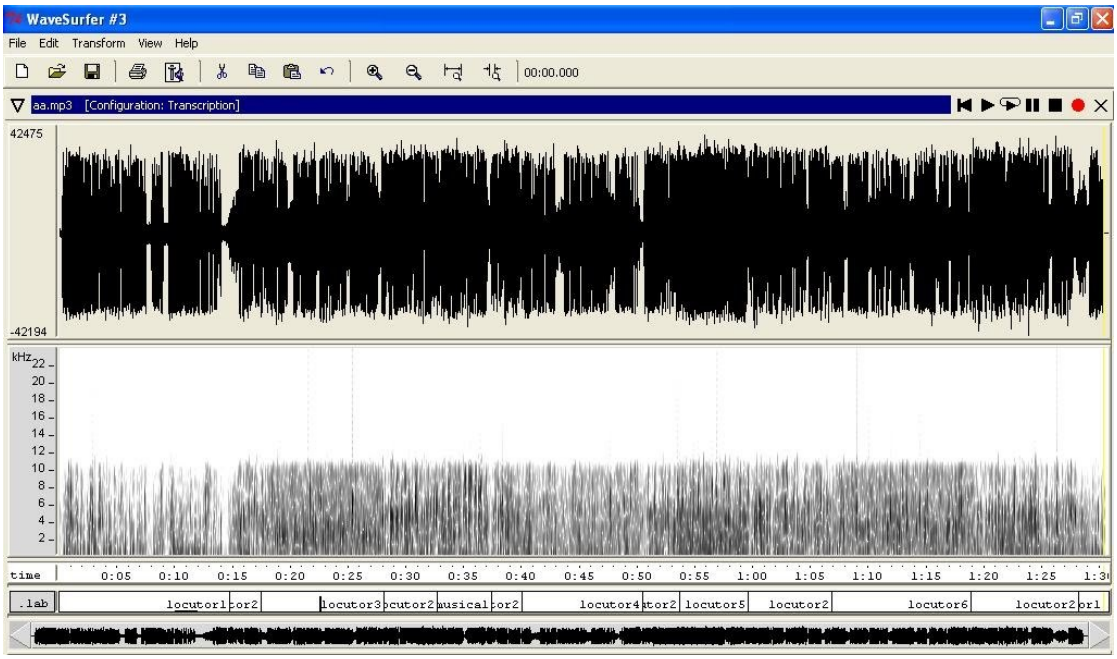


Figura 3.2. Wavesurfer un cop carregats els panells.

```
0.0000000 14.7958026 locutor1
14.7958026 17.4557221 locutor2
17.4557221 28.0954004 locutor3
28.0954004 32.7502596 locutor2
32.7502596 37.4051188 musica1
37.4051188 40.0650384 locutor2
40.0650384 50.5384717 locutor4
50.5384717 53.6971261 locutor2
53.6971261 59.6209142 locutor5
59.6209142 66.8619400 locutor2
66.8619400 78.8989700 locutor6
78.8989700 88.2088604 locutor2
88.2088604 90.3717642 locutor1
```

Figura 3.3. Tros d'un fitxer .lab

A més a més, wavesurfer permet definir etiquetes per defecte amb la finalitat d'estalviar el temps d'escriure-les.

3.4. MPEG-7 Audio Encoder

MPEG-7 Audio Encoder [20] és una llibreria de Java que permet la descripció de contingut d'àudio utilitzant alguns dels descriptors definits per l'estàndard MPEG-7. Va ser desenvolupada al Institute of Communications Enginering [30], el qual pertany a la universitat Aachen University (RWTH) – Germany [31] sota llicència LGPL (GNU Lesser General Public License).

El contingut de la llibreria es presenta de manera gràfica mitjançant una GUI on es permet carregar un fitxer d'àudio i veure els resultats per pantalla o bé guardar l'arxiu .xml resultant, o bé es pot executar a partir de línia de comandes.

```
Java -jar MPEG7AudioEnc.jar [path/]audio{.wav|.au|.aiff} > audio.MPEG-7
```

Els descriptors que permet extreure el codificador són els següents són gairebé tots els definits a l'estàndard.

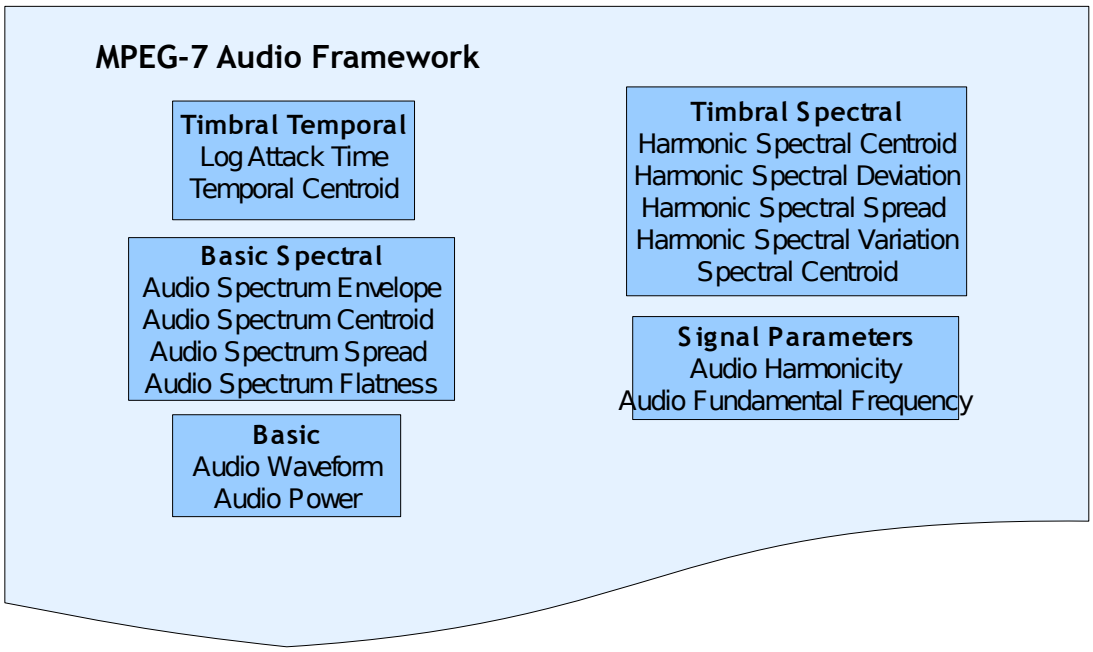


Figura 3.4. Descriptors de baix nivell que permet seleccionar MPEG-7 Audio Encoder

En el cas de no voler utilitzar tots els descriptors que es presenten a la llibreria, es pot passar fitxer de configuració com un altre paràmetre en cas de treballar des de comandes, o seleccionar els descriptors que volem a partir de la interfície gràfica i guardar l'esquema de descripció.

```
<?xml version="1.0" encoding="UTF-8"?>
<Config
  xmlns="http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd"
  xmlns:mp7ae="http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd
        http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd">

<!-- set hop size for all modules -->
<Module xsi:type="Resizer">
  <HopSize>10</HopSize>
</Module>

<!-- Enable AudioWaveformType -->
<Module xsi:type="AudioWaveform" mp7ae:enable="true" />

<!-- Enable AudioSpectrumEnvelope with loEdge=62.5Hz, hiEdge=16kHz and ocave resolution=1.0 -->
<Module xsi:type="AudioSpectrumEnvelope" mp7ae:enable="true">
  <loEdge>62.5</loEdge>
  <hiEdge>16000.0</hiEdge>
  <resolution>1.0</resolution>
</Module>

</Config>
```

Figura 3.5. Exemple fitxer configuració

A més, es permeten modificar alguns paràmetres generals pel que fa a la generació dels descriptors, com interns de cada descriptor.:

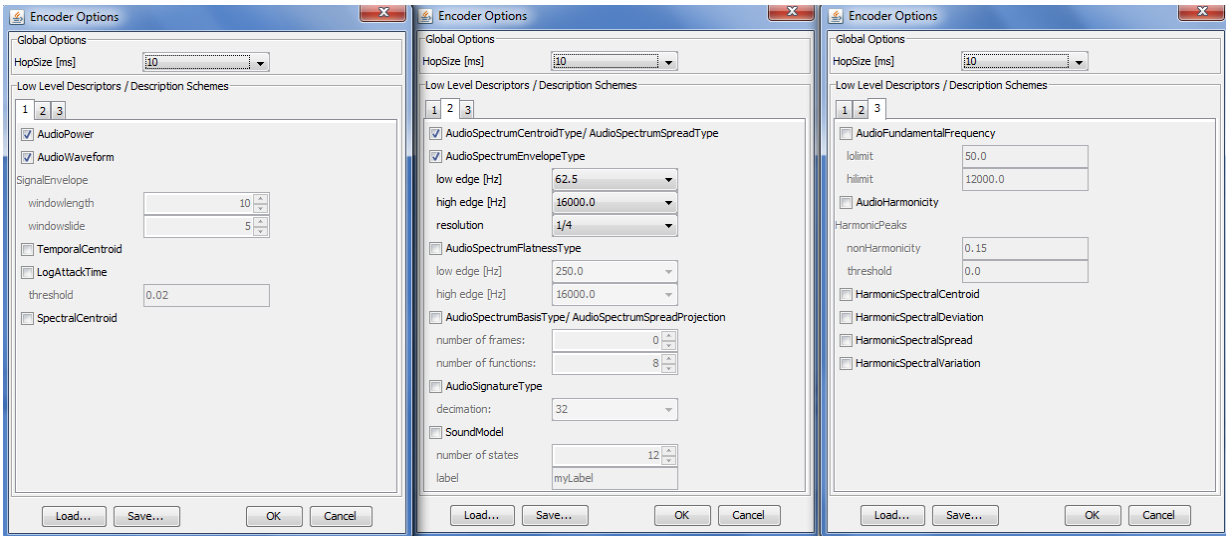


Figura 3.6. Interfície MPEG-7 Audio Encoder per modificar paràmetres

Com es pot veure a la figura 3.5, els paràmetres que es permet modificar són molts, i tots definits per l'estàndard MPEG-7. A més també es permet modificar la finestra que s'utilitzarà per calcular els descriptors a partir del paràmetre *hopsize,* els valors el qual pot obtenir són 10ms o 30ms.

3.5 XMLBeans

XML Beans [32], és una tecnologia per tal d'accedir a les dades emmagatzemades a un fitxer xml mitjançant classes de java. Així doncs mitjançant aquestes classes podrem llegir i escriure arxius xml amb força rapidesa i facilitat.

3.6 Weka

Un Weka o woodhen (Gallirallus) pertany a la família de les aus, i és una espècie pròpia de Nova Zelanda. És per això que els investigadors de la universitat de Waikato (Nova Zelanda) [33wiki] van anomenar al seu projecte de la mateixa manera: Weka[33].

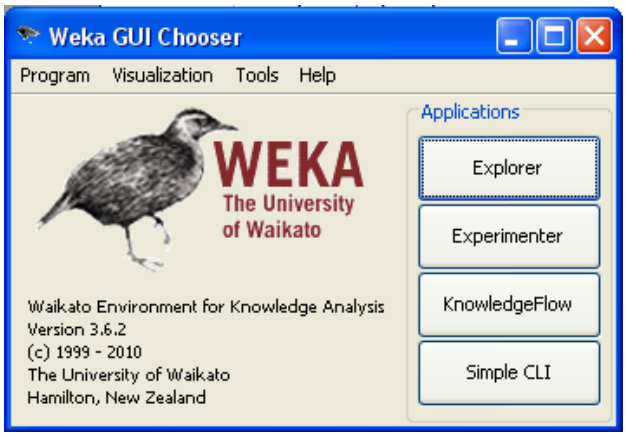


Figura 3.7. Pagina presentació Weka

El Weka [34][35] és un programa de data mining, és a dir, mineria de dades. La mineria de dades consisteix en l'extracció d'informació no trivial que resideix de manera no implícita a les dades. A més, aquests programes gestionen les dades, i permeten realitzar tasques de predicció, classificació i segmentació. El software està format per un conjunt de llibreries JAVA que permeten tractar volums molt grans de dades i crear de manera automàtica models per descriure aquestes dades les quals serien molt difícil d'analitzar de manera manual. Així doncs estem parlant en tot moment de l'aprenentatge automàtic en el món de les computadores.

Pel que fa al 'interacció amb el programa, hi ha diferents possibilitats seleccionables al menú principal: Explorer, Experimenter, KnowedgeFlow i per últim SimpleCLI. Cadascuna d'aquestes possibilitats són en certa manera, independents de les altres. Això vol dir que el seu desenvolupament és en part independent entre les quatre. En conseqüència no totes les modalitats tenen exactament les mateixes funcionalitats ni estan orientades pel mateix ús, sense perdre de vista que és el mateix programa i que totes les opcions tenen el mateix fi.

L'opció Explorer és la més interactiva, amb una interfície gràfica que permet seleccionar visualment les funcionalitats. Així doncs, el programa permet carregar dades, tractar-les a partir de filtres de pre-processat, visualitzar-les de diferents maneres i finalment, crear els models i classificar les dades.

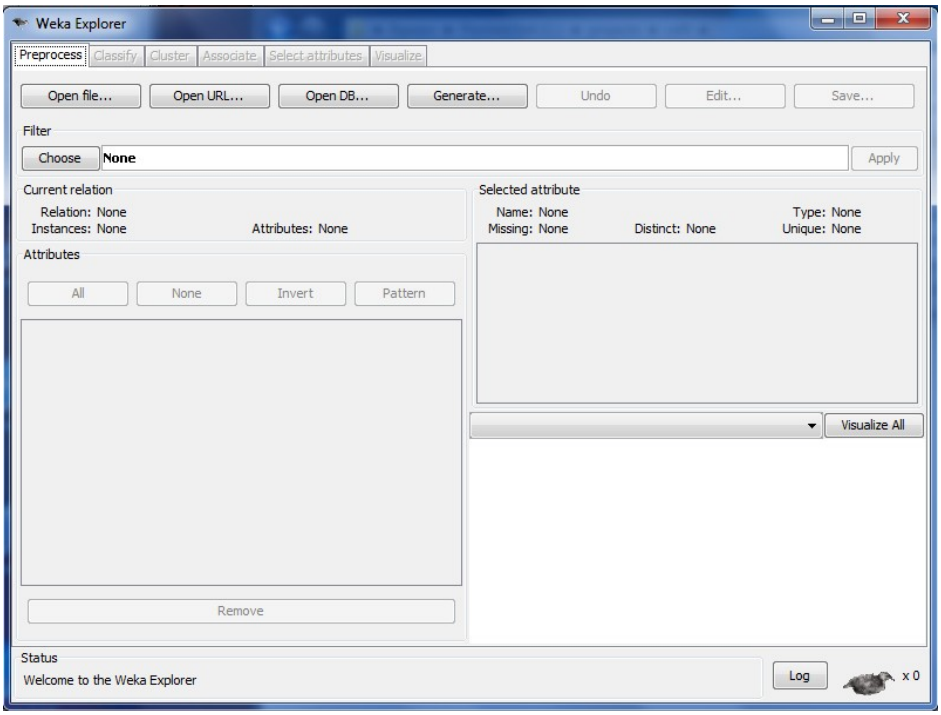


Figura 3.8. Weka en mode Explorer

Pel que fa a la modalitat Explorer, tal i com es pot veure a la figura 3.5, trobem sis pestanyes a la part superior: Preprocess, Classify, Cluster, Associate, Select attributes, Visualize. A la primera de totes, Prepocess és on es permet carregar les dades d'entrada, seleccionar les que ens interessen i aplicar diferents filtres. En quant als formats d'entrada de les dades, Weka accepta els següents formats: arff, C4.5, Csv... entre d'altres.

El format propi de Weka és arff, acrònim de *Attribute-Relation File Format*, el qual ha de presentar una estructura determinada per que el programa pugui interpretar de manera correcta les dades. Aquest format es pot tractar com un fitxer de text comú i consta de les següents parts diferenciades i representades en l'ordre en que es presenten:

- Capçalera: És on es defineix el nom de la relació, tanmateix, se pot considerar com el nom de la classificació. La sintaxi és la següent.

```
@relation <nom-de-la-relació>
```

On <nom-de-la-relació> és del tipus String i no admet espais.

- Declaració dels atributs: Aquí es declaren els atributs de la nostra relació juntament amb el tipus de dades.

```
@attributte <nom-atribut> <tipus-atribut>
```

<nom-atribut> és de tipus String i no admet espais. Weka defineix diferents tipus d'atribut:

  - NUMERIC: Expressa nombres reals.
  - INTEGER: Expressa nombres enters.
  - DATE: Expressa dates.
  - STRING: Expressa cadenes de caràcters.
  - ENUMERATE: Expressa un conjunt de possibles valors finits.

```
@attribute color_ulls {blau, marró, mel, verd, negre, gris}
@attribute edat numeric
```

- Dades: És on pròpiament s'escriuen les dades. La manera d'escriure-les ha de ser conseqüent a cóm els atributs han estat definits. Així doncs, el format serà el següent:

```
@data
nena, blau, 15, 1.46, Cerdanyola del Vallès
nen, marró, 18, 1.78, Barcelona
nen, marró, 21, 1.68, Terrassa
nena, verd, 14, 1.70, Barcelona
nena, mel, 19, 1.65, Rubí
```

Així doncs, un fitxer arff ha de tenir de manera ordenada els 3 components que s'han presentat anteriorment. Un exemple és la relació entre els nets de la senyora Carme, on figura si son nens o nenes, amb els atributs de color d'ulls, edat, alçada i localitat a la que viuen.

```
@relation nets_de_la_Carme

@attribute label {nen,nena}
@attribute color_ulls {blau, verd, marró}
@attribute edat numeric
@attribute alçada numeric
@attribute localitat string

@data
nena, blau, 15, 1.46, Cerdanyola del Vallès
nen, marró, 18, 1.78, Barcelona
nen, marró, 21, 1.68, Terrassa
nena, verd, 14, 1.70, Barcelona
nena, mel, 19, 1.65, Rubí
```

Un cop les dades es carreguen es disposen de diferents eines per tractar aquestes o per visualitzar-les. En cas de voler visualitzar-les, directament hem d'anar a la pestanya Visualize i ha d'aparèixer un menú amb les gràfiques que relacionen els diferents atributs, amb diferents opcions pel que fa a la manera de mostrar aquesta informació.

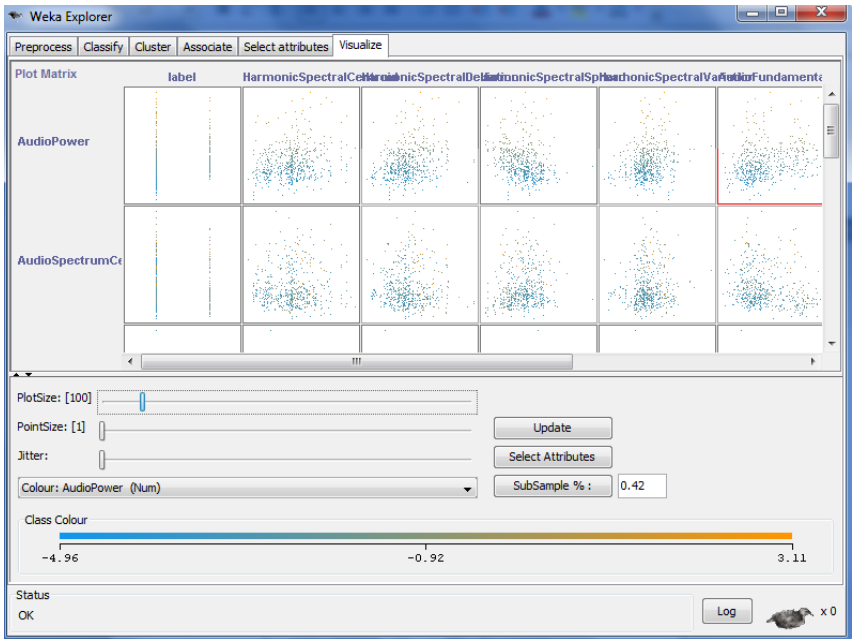


Figura 3.9. Pantalla Weka visualització de les dades

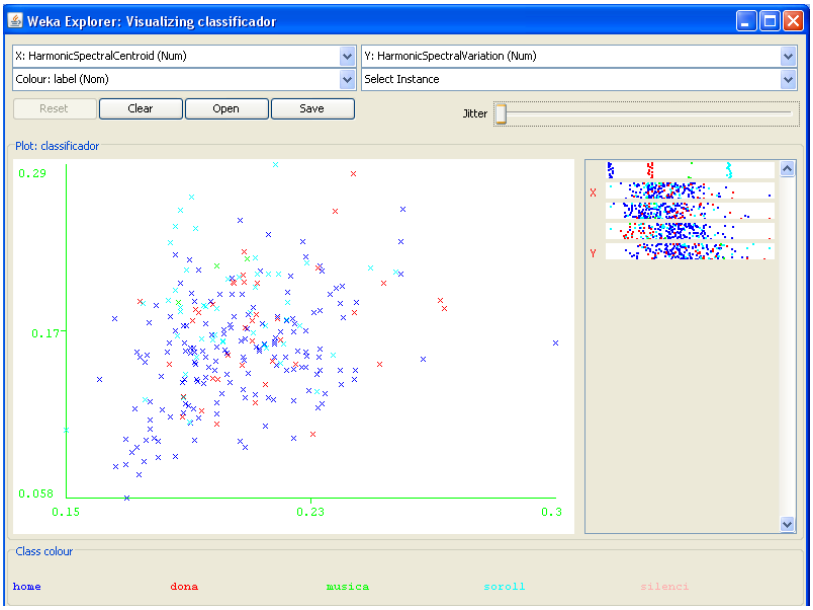


Figura 3.10. Augment de visualització de les dades en funció de dos paràmetres

Pel que fa al procés de classificació, hem d'anar a la pestanya classify. Apareixen diferents opcions, les més importants són Choose Classifier, que ens permet seleccionar l'algorisme de classificació, Test Options, on es permet seleccionar totes les dades per realitzar el classificador, carregar un altre arxiu de test o bé seleccionar un tant per cent corresponent a les dades d'entrada inicials destinat a testejar automàticament els resultats, i per últim seleccionar l'atribut en funció del qual es vol realitzar la classificació.

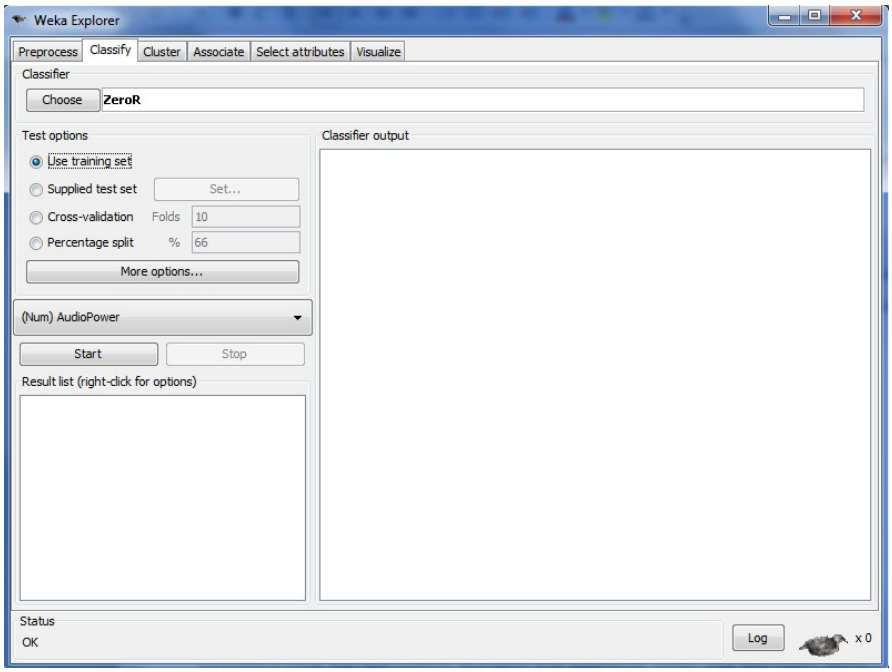


Figura 3.11. Pantalla Weka de classificació

Els classificadors que incorpora Weka es troben dividits en carpetes depenen de la seva implementació. Dins de cada carpeta es troben tots els classificadors. Si es selecciona a sobre del classificador queda automàticament seleccionat. A més, alguns classificadors permeten modificar opcions internes, aquestes opcions es fan visibles fent clic a sobre del mateix nom del classificadors**.**

Per tal de visualitzar els resultats de classificació, hi ha dues maneres de fer-ho. La primera i la més visible, és visualitzar els resultats que surten per pantalla. Aquests resultats depenen del classificador, però en general, es poden visualitzar una sèrie d'estadístiques pel que fa al test de classificació, una matriu de confusió, i per últim les regles seguides alhora de classificar.

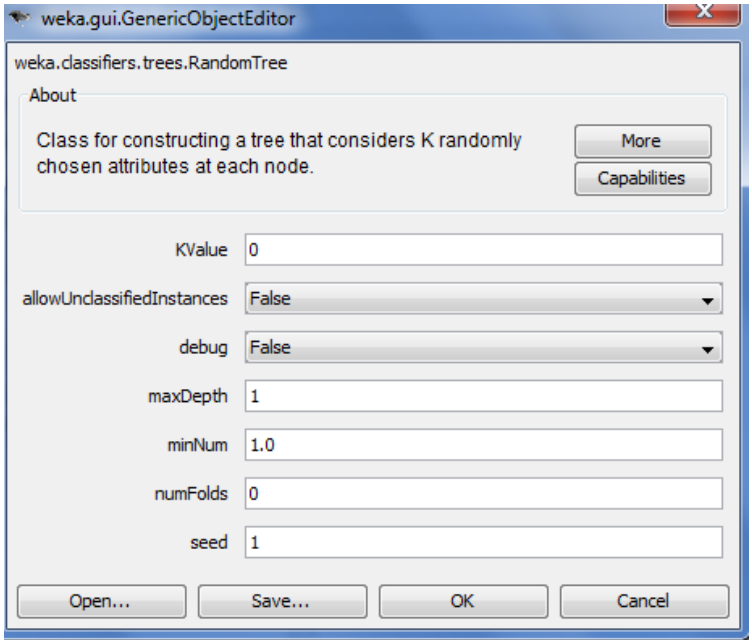


Figura 3.12. Opcions internes del classificador RandomTree

```
Time taken to build model: 0.02 seconds

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          58               75.3247 %
Incorrectly Classified Instances        19               24.6753 %
Kappa statistic                          0.4555
Mean absolute error                      0.1456
Root mean squared error                  0.2816
Relative absolute error                 67.3499 %
Root relative squared error             86.2682 %
Total Number of Instances               77
```

Figura 3.13. Estadístiques del test segons la classificació

```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
 48  0  1  0  0 |  a = home
  5  7  0  0  0 |  b = dona
  1  0  0  0  0 |  c = musica
  9  2  1  3  0 |  d = soroll
  0  0  0  0  0 |  e = silenci
```

Figura 3.14 Matriu de confusió

Pel que fa a la visualització de les regles seguides a l'hora de classificar, es pot visualitzar a més a més a partir de l'opció visualize tree, al fer clic a sobre del classificador generat. I per últim totes les dades que han aparegut per pantalla es poden guardar a partir de l'opció Save Buffer.

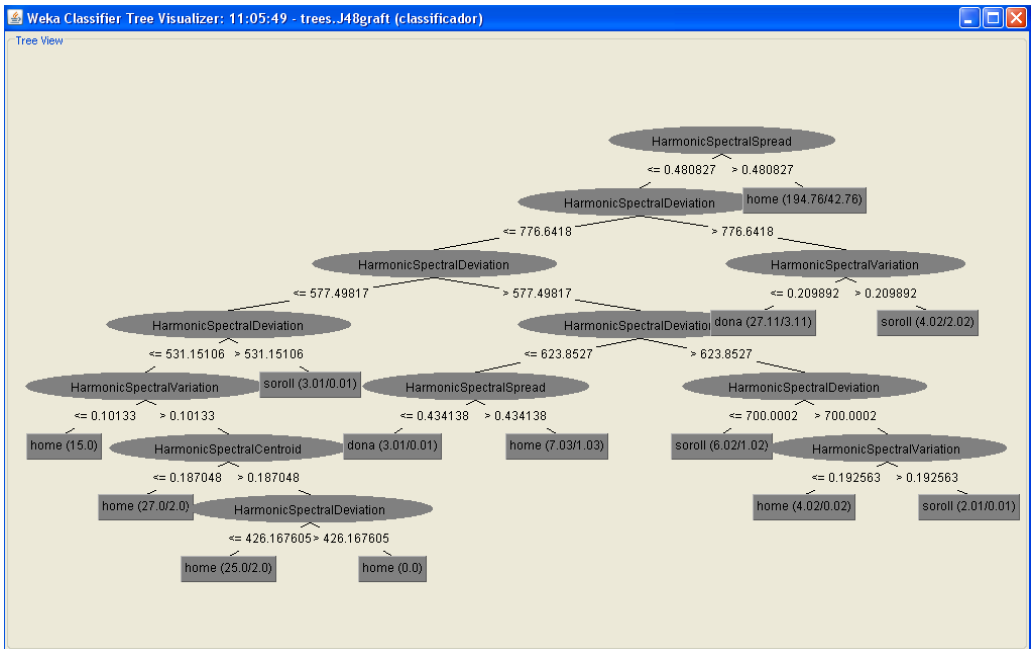


Figura 3.15. Arbre de classificació.

4. Base de dades

La base de dades que s'ha generat per tal d'entrenar els models i que s'utilitzarà posteriorment per comprovar els resultats es composa de tres programes de continguts diferents : tertúlies, entrevistes i reportatges. Aquesta varietat d'estils permetran després treure millors conclusions sobre el comportament del classificador en diferents situacions.

En primer lloc, tenim dues edicions del programa "A banda i banda", d'una duració d'uns 50 minuts aproximadament cadascun. El format del programa consisteix en una petita presentació dels temes a tractar, una entrevista-reportatge i el posterior debat. Per acabar tornen a l'entrevista i acaben de debatre-la. Per tant presenta un alt contingut en locutors.

En segon lloc, una edició del programa "Set Dies", amb una duració d'uns 50 minuts. Aquest programa també presenta molt contingut de veus ja que és un programa d'entrevista. L'estructura és de dues o tres entrevistes per programa. D'aquesta manera apareixen només entrevistador i entrevistat.

Per últim, "Naturalment" és un programa d'entrevistes i reportatges sovint a l'aire lliure, la qual cosa fa que hi hagi molta barreja de sons, amb soroll i /o músiques de fons. En aquest cas es disposa de dues edicions del programa, d'una duració d'uns 50 minuts cadascuna. A la figura 4.1 es pot veure els detalls dels diferents vídeos seleccionats.

Nom al servidor	Canal	Data emissió	Nom arxiu	Duració
A BANDA I BANDA 11_230110	El 9TV (XTL)	23/01/10	banda1	00:52:16
A BANDA I BANDA 12_300110	El 9TV (XTL)	30/01/10	banda2	00:49:39
NATURALMENT 08	El 9TV (XTL)	???/???/???	naturalment8	00:49:33
NATURALMENT 10	El 9TV (XTL)	???/???/???	naturalment10	00:49:54
- 7 DIES 1 260310 - 7 DIES 2 260310 - 7 DIES 3 260310	EL 9TV (XTL)	26/03/10	setdies	00:54:25
TOTAL				**4h 27min 47s**

Figura 4.1 Taula dels continguts de la base de dades

Tots ells provenen del canal EL 9TV, un canal municipal client de VSN. Els arxius han sigut descarregats del servidor privat de l'empresa on es troben tots els vídeos dels diferents clients, amb el consentiment d' EL 9TV d'utilització d'aquests per al desenvolupament del projecte.

4.1 Procés de segmentació manual i descripció de la base de dades.

La creació de la base de dades només contempla obtenir els arxius de vídeo que la conformen. S'ha de fer tot un procés per tal que aquesta informació sigui aprofitable per a poder dissenyar i avaluar el sistema que es vol crear. Per tant, es necessari fer primer una descripció manual del contingut d'aquests vídeos. Aquest procés es coneix com segmentació manual. Un cop feta pot començar l'extracció de les conclusions més significatives sobre el contingut de la base, i sobretot, un segon procés d'extracció de la informació de cada segment.

Aquest primer procés d'adequació de la base de dades esta representat en la figura 4.2, on es veu els diferents passos que hi intervenen.

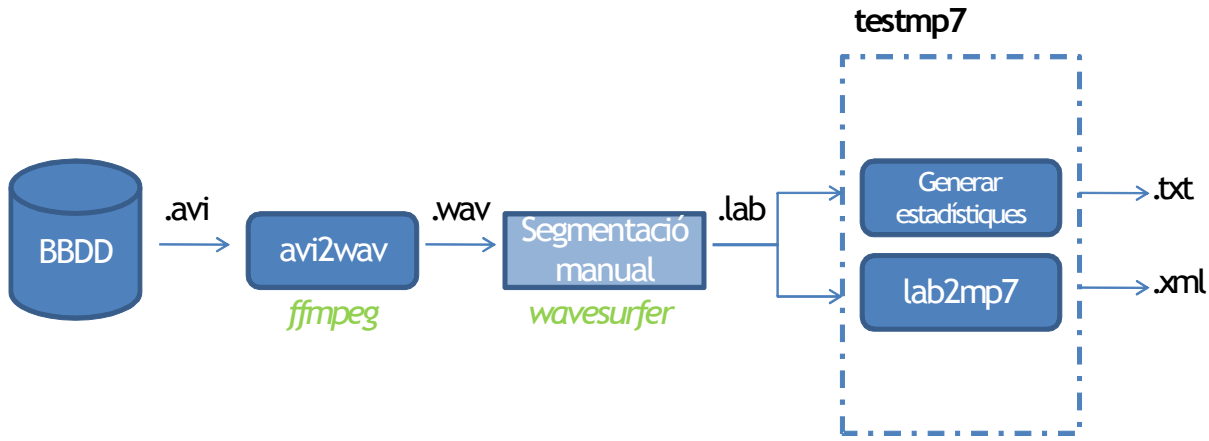
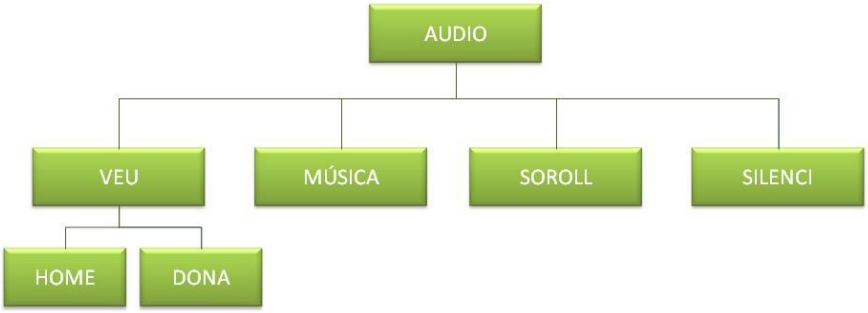


Figura 4.2 Diagrama del procés de creació de la BBDD

Com es pot observar, el procés es semiautomàtic ja que com es comprensible la segmentació manual s'ha de portar a terme per una persona. A continuació s'expliquen amb més detall tots el passos que conté aquest procés.

4.1.1 Segmentació manual i criteris d'etiquetat

La segmentació manual requereix de molta cura i de la màxima precisió, ja que els resultats obtinguts finalment poden veure's molt afectats depenent de si s'ha fet correctament o no. A més, existeix el problema que és un procés subjectiu, que depèn de qui ho fa, i essent dues persones les que ho fem, segurament els resultats variïn segons qui ho fa. Cal doncs un pas previ on definir amb de la forma més exacte possible les diferents etiquetes a utilitzar, les barreres entre aquests, i tot un seguit de normes per tal de que la segmentació sigui precisa i uniforme independentment de qui la hagi fet. Aquesta unificació de criteris, s'ha fet conjuntament amb el director del projecte, buscant que la sortida sigui similar a la que esperem que tregui el nostre classificador. Així doncs, s'ha creat un arbre, representat a la figura 4.3 , on es mostren les diferents etiquetes a utilitzar.



4.3. Criteris d'etiquetatge i segmentació manual.

La classificació final serà per tant entre 5 categories(home, dona,música,soroll,silenci) que s'han anomenat "pures". No obstant, qualsevol arxiu conté barreges d'aquestes categories, de forma que apareixen dues noves categories: veu+musica (homemusica i donamusica) i veu+soroll (homesoroll i donasoroll). Finalment, les etiquetes que s'utilitzaran són les següents:

1) **homeN/donaN:** cada cop que aparegui un home/dona nou incrementarem N començant desde 1 fins al número total d'homes/dones que apareguin. Aquesta etiqueta nomes s'assignarà en cas de tenir una veu masculina/femenina pura.

2) homeNmusica, donaNmusica: aquesta etiqueta s'assignarà en qualsevol cas en què hi hagi una veu juntament amb un element sonor musical. Quan una musica es perceptible a la oïda però es pot comprendre el missatge de la veu, es dona aquesta etiqueta.

3) **homeNsoroll, donaNsoroll**: aquesta etiqueta s'assignarà en tots els casos als quals hi hagi una veu per sobre d'un soroll, el qual en cap cas ha de dificultar l'enteniment del contingut del que està dient el locutor. També ens podem guiar per altres factors com ara presencia de freqüències a l'espectre per sobre dels 5KHz o un augment considerable en el nivell de pressió sonora que es pot observar a la variació temporal del senyal.

4) **Soroll**: Em decidit que etiquetarem com a soroll tot allò que no sabem que és. Exemples poden ser: soroll convencional (blanc, rosa), solapament de locutors, soroll amb veu molt elevat, varis...

5) **Silenci:** El que entenem per silenci major a tSilenci=2 segons. D'aquesta manera evitem que petits talls (com poden ser agafar aire, un sospir... d'un locutor), tallin el contingut.

6) Música: Només en el cas de que existeixi música sense cap altre tipus de soroll o veu.

Aquestes etiquetes han sigut assignades manualment utilitzant el programa wavesurfer, de la forma que ja hem explicat en capítols anteriors, i per fer-ho s'han utilitzat els arxius d'àudio (en format .wav) extrets del vídeo mitjançant un algoritme que crida a un altre programa del qual hem parlat abans, el ffmpeg, que es el que fa l'extracció segons les comandes ja exposades en el capítol 3.

4.1.2 testMPEG-7

Els arxius de segmentació manual (.lab) els introduïm en un algorisme que hem anomenat *TestMPEG-7*. El nom prové del fet que el vam començar com una proba per veure com es podia llegir arxius MPEG-7 (xml) a partir de llibreries, però hem anant introduint-li millores i ara per ara fa tres funcions, dues de les quals tenen a veure amb aquest apartat i de les que parlarem a continuació. L'altra funció l'explicarem en un pròxim apartat. Les dues que explicarem a continuació i que es trobem representades a la FIGURA 4.2 són l'escriptura d'un arxiu de text (.txt) detallant d'informació de cada programa i l'escriptura d'un arxiu de descripció del programa sota l'estàndard MPEG-7.

Per una banda, l'obtenció d'un arxiu d'estadístiques en forma .txt no comporta cap complicació. No obstant, en primera instància per tal de construir els models ja s'ha explicat que es volen només els arxius purs. Per això la sortida de les estadístiques avalua com a soroll tots aquells segments que no siguin purs. D'aquesta manera es pot veure quins programes són més indicats per a ser utilitzats a la fase d'entrenament. En l'arxiu d'estadístiques obtenim els percentatges de temps de cada etiqueta.

Per una altra banda, generar un arxiu de descripció del contingut XML seguint l'esquema MPEG-7 comporta més complexitat a l'hora de programar. S'ha utilitzat la llibreria creada a partir d' XMLBeans per tal de poder anar escrivint el document, seguint l'esquema presentat en el capítol 2. També el parsejat, que comprova la correcció del document final, l'obtenim gràcies a la llibreria anteriorment mencionada.

```
- <Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <Description xsi:type="urn:ContentEntityType" xmlns:urn="urn:mpeg:mpeg7:schema:2001">
    - <MultimediaContent xsi:type="urn:AudioType">
      - <Audio>
        - <MediaTime>
            <MediaTimePoint>0.0</MediaTimePoint>
            <MediaDuration>0.0</MediaDuration>
          </MediaTime>
        - <TemporalDecomposition>
          - <AudioSegment id="0">
            - <TextAnnotation>
                <FreeTextAnnotation>silenci</FreeTextAnnotation>
              </TextAnnotation>
            - <MediaTime>
                <MediaTimePoint>0.0</MediaTimePoint>
                <MediaDuration>3.01</MediaDuration>
              </MediaTime>
            </AudioSegment>
          </TemporalDecomposition>
        - <TemporalDecomposition>
          - <AudioSegment id="1">
            - <TextAnnotation>
                <FreeTextAnnotation>musica1</FreeTextAnnotation>
              </TextAnnotation>
            - <MediaTime>
                <MediaTimePoint>3.01</MediaTimePoint>
                <MediaDuration>20.935125</MediaDuration>
              </MediaTime>
            </AudioSegment>
          </TemporalDecomposition>
        - <TemporalDecomposition>
```

Figura 4.4. Exemple d'un document MPEG-7 de sortida de l'arxiu banda2

Com es veu a la figura 4.4, després de declarar la capçalera el que fem es declarar una descripció del contingut(especificant que el contingut es àudio) una descomposició temporal. Creem un àudio segment, li assignem un numero com a identificador, seguit un ordre d'aparició, i per a donar-li l'etiqueta, li afegim al segment una anotació de text. També utilitzem els indicadors propis de MPEG-7, MediaTimePoint i MediaDuration, par tal d'indicar el temps d'inici i la duració de cada segment.

4.2 Estadístiques

A continuació es detalla la informació obtinguda dels arxius d'estadístiques, a més de fer una valoració sobre el seu contingut. S'ha separat per tipus de programa, per tal de veure més clar les característiques de cadascun d'ells.

4.2.1 "A banda i banda"

Aquests tipus de programes aporten un alt contingut en veus, principalment veus masculines, ja que els tertulians són en majoria homes. El contingut en música es bastant baix degut a que moltes de les sintonies que sonen es troben acompanyades de veu i per tant, les descartarem. El contingut en soroll també és força elevat degut a que hi ha molts elements sonors que no els podem identificar, i sobretot molta superposició entre veus i els classificarem a la classe per defecte definida com "soroll".

Els pocs moments de VeuMusica són la principi i al final, quan es solament la sintonia d'entrada i sortida amb la presentadora

Banda 1

Etiqueta	Duració (segons)
Veu	2900.01
Música	41.84
Silenci	0.0
Soroll	175.61
TOTAL	**3120.77**

Etiqueta	Duració (segons)
Dones	376.78
Homes	2523.23

Etiqueta	Duració (segons)
VeuMusiva	1,71
VeuSoroll	0
Soroll pur	173,9

Figura 4.5 Taules amb les dades de Banda1

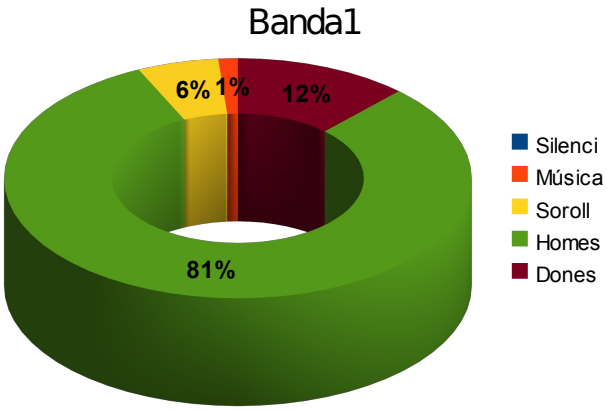


Figura 4.6 Gràfica toroïdal amb els percentatges d'aparició de cada etiqueta principal a Banda1

Banda 2

Etiqueta	Duració (segons)
Veu	2749.55
Música	45746
Silenci	11517
Soroll	168.92
TOTAL	**2975730**

Etiqueta	Duració (segons)
Dones	648,68
Homes	2100,87

Etiqueta	Duració (segons)
VeuMusiva	3,7
VeuSoroll	0
Soroll pur	165,2

Figura 4.7 Taules amb les dades de Banda2

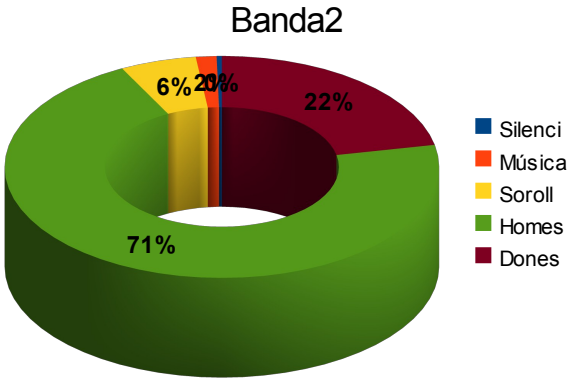


Figura 4.8 Gràfica toroïdal amb els percentatges d'aparició de cada etiqueta principal a Banda2

4.2.2 "Set dies"

El contingut d'aquest programa es semblant al de banda i banda, ja que tots dos tenen un fort contingut de veus, i els dos tenen, en molta més proporció, veus d'homes. Potser l'única diferencia es el menor nombre de locutors en set dies, degut a que son converses entre presentador i entrevistat.

Etiqueta	Duració (segons)
Veu	2947.80
Música	72653
Silenci	0.0
Soroll	199.99
TOTAL	**3265.60**

Etiqueta	Duració (segons)
Dones	585.28
Homes	2362.52

Etiqueta	Duració (segons)
VeuMusiva	112.42
VeuSoroll	42,54
Soroll pur	45,03

Figura 4.9 Taules amb les dades de set dies

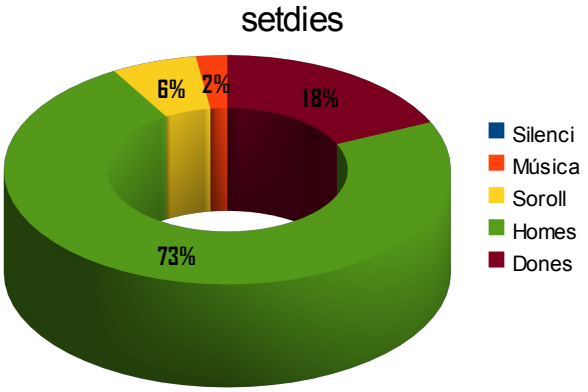


Figura 4.10 Gràfica toroïdal amb els percentatges d'aparició de cada etiqueta principal a set dies

4.2.2 Naturalment

Aquests programes de reportatges no semblen aportar gaire contingut per a l'entrenament dels models, almenys per veus. Tenen, això si, un major contingut en música que els anteriors. Com es pot observar, son en gran majoria soroll, però com es reflexa a les gràfiques de detall del soroll, la gran majoria es degut a la superposició de la veu sobre música o soroll, aquest degut a que els reportatges es porten a terme en exteriors i per tant hi ha forçosament soroll de fons.

Naturalment 8

Etiqueta	Duració (segons)
Veu	483.01
Música	150.95
Silenci	**101.86**
Soroll	1957.78
TOTAL	**2973.89**

Etiqueta	Duració (segons)
Dones	96.76
Homes	386.25

Etiqueta	Duració (segons)
VeuMusiva	500.89
VeuSoroll	1108,05
Soroll pur	348,89

Figura 4.11 Taules amb les dades de Naturalment8

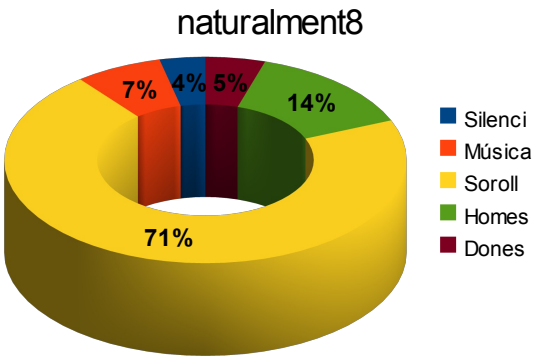


Figura 4.12 Gràfica toroïdal amb els percentatges d'aparició de cada etiqueta principal a Naturalment 8

Naturalment 10

Etiqueta	Duració (segons)
Veu	99.44
Música	715.87
Silenci	144.82
Soroll	2030.32
TOTAL	**2993.79**

Etiqueta	Duració (segons)
Dones	15,02
Homes	84.23

Etiqueta	Duració (segons)
VeuMusiva	692.5052
VeuSoroll	10155.59
Soroll pur	322,22

Figura 4.13 Taules amb les dades de Naturalment8

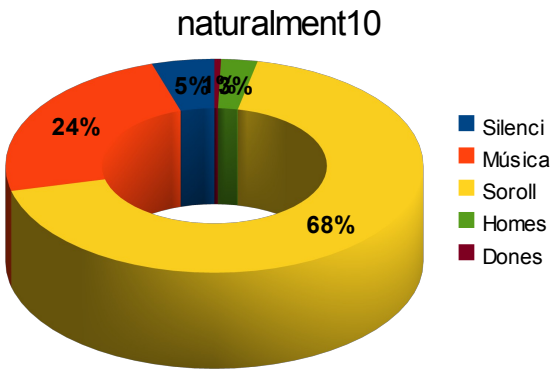


Figura 4.14 Gràfica toroïdal amb els percentatges d'aparició de cada etiqueta principal a Naturalment 8

5. Sistema de classificació

Un cop tenim la base de dades ja creada, el següent pas correspon al procés de creació del sistema de classificació. Independentment de les eines que s'utilitzin per realitzar aquest procés la manera de procedir en tots els casos estudiats és molt semblant. Concretament ens basarem en el procediment seguit a [12].

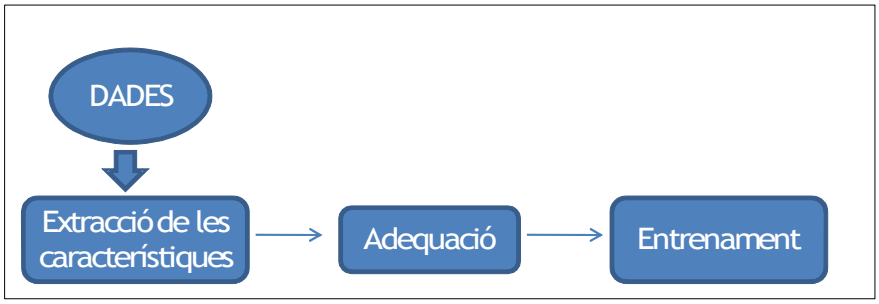


Figura 5.1. Esquema genèric del procés d'entrenament

Tal i com es pot apreciara la Figura 5.1, el procés en general consta de 3 passos: extracció de les característiques, adequació de les dades obtingudes i finalment l'entrenament pròpiament dit. Al final d'aquestes 3 etapes, s'obtenen una sèrie de models entrenats en funció de les dades d'entrada.

En aquest punt, és important disposar d'una bona base de dades per garantir un procés d'entrenament correcte. Una bona base de dades no sempre queda determinada pel volum de dades que es disposa si no també, per la qualitat de les dades les quals han estat sotmeses a un procés de segmentació i etiquetat manual, és a dir, possibles errors humans, per la qualitat de l'obtenció d'aquestes que en cada cas serà diferent, i quant més genèric es vol el classificador, més ventall de dades diferents s'ha de disposar. Tots aquests requeriments juntament amb un volum elevat de dades, constitueix una base de dades homogènia i no susceptible a possibles errors deguts al sobre-entrenament de les dades.

5.1. Extracció i adequació de les característiques de les dades

En conseqüència, el primer pas és l'extracció de les característiques que resideixen de manera implícita als arxius sonors.

L'extracció de les característiques de l'àudio es pot realitzar de diferents maneres i utilitzant diferents algorismes. En el cas del projecte que presentem, les dades ens van arribar de manera compacta en cinc programes televisius diferents. Per tal d'extreure les característiques de tots els segments, primer de tot s'ha de disposar de les bases de dades secundàries, o sigui, un conjunt de segments sonors corresponents a una classe. Així doncs, el primer pas serà, combinant la segmentació manual amb els arxius sonors, generar porcions d'àudio segons marca l'arxiu de segmentació .lab. Aquest procés queda contemplat al programa testmp7 al qual resideixen altres funcionalitats ja explicades anteriorment.

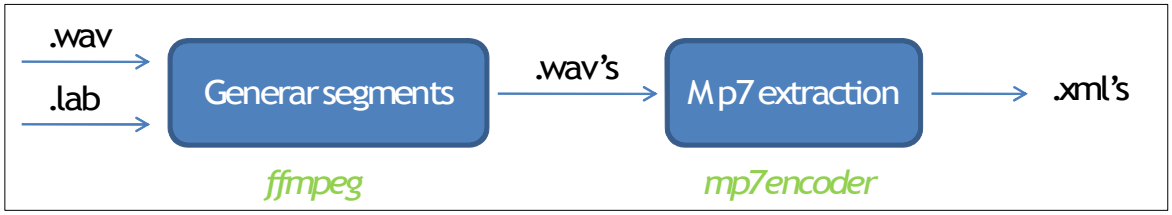


Figura 5.2. Esquema generació de descripció.

D'aquesta manera el resultat final seran quatre grans bases de dades de música, soroll, silenci i veu (on es contemplaran diferències entre home i dona) formades per petits segments extrets de la base de dades principal. Els arxius de sortida del mòdul Generat segments, segueixen la següent sintaxi:

```
nomdel'arxiu_n°segment_etiqueta.wav
nomdel'arxiu_n°segment_etiqueta.xml
```

Com a exemple, per l'arxiu de vídeo banda1, tindrem els arxius que es mostren a la figura 5.3:

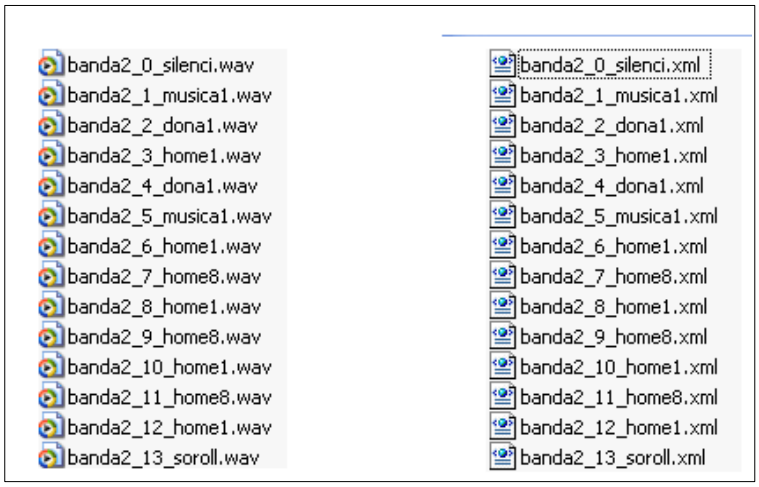


Figura 5.3. Exemple d'assignació de noms.

Tal i com s'ha detallat a l'apartat 3, l'extracció de la informació implícita a l'àudio, es realitzarà a partir dels descriptors de baix nivell definits per l'estàndard mpeg-7[4] i implementats a les llibreries Java Mpeg-7 Audio Encoder[19]. La decisió de les eines a utilitzar per a l'extracció de les característiques no va ser quelcom trivial, va estar relacionada amb el procés de recerca desenvolupat al llarg de tot el projecte. Cal dir que la selecció dels descriptors adequats pot ser molt decisiva en quant a l'èxit o el fracàs d'un classificador.

Així doncs, pel que fa a l'extracció de les característiques, és a dir, els descriptors, existeix gran varietat d'algorismes a part dels definits per l'estàndard Mpeg-7 (low-level descriptors) i moltes vegades més utilitzats que aquests. Com ja es parla a l'apartat 1.1, alguns dels més utilitzats pel que fa al reconeixement de veu són Mel-scale Frequency Cepstrum Coefficients (MFCC) [36], la Transformada ràpida de Fourier (FFT), detecció de pitch (to) o Freqüència Fonamental, Autocorrelació, Predicció lineal (LPC), Número de encreuaments per zero (ZCR), etc. Tal i com s'exposa a l'apartat 1.1 en alguns casos, l'elecció dels descriptors esdevé una barreja entre descriptors mpeg-7 i diferents algorismes també útils com els que hem exposat anteriorment, com s'exposa a [12], la utilització de descriptors totalment independents als descrits a mpeg-7 [7], [37] o, per contra, la utilització única de descriptors mpeg-7 [11], [15].

Finalment, la decisió es va basar en els estudis referenciats anteriorment i en la idea inicial de realitzar un classificador basat en els descriptors de baix nivell descrits a l'estàndard MPEG-7. Un cop seleccionat el mètode i les eines que s'utilitzaran per extreure les característiques de l'àudio, procedirem a implementar l'esmentat procés de la manera que es mostra a la Figura 5.4.

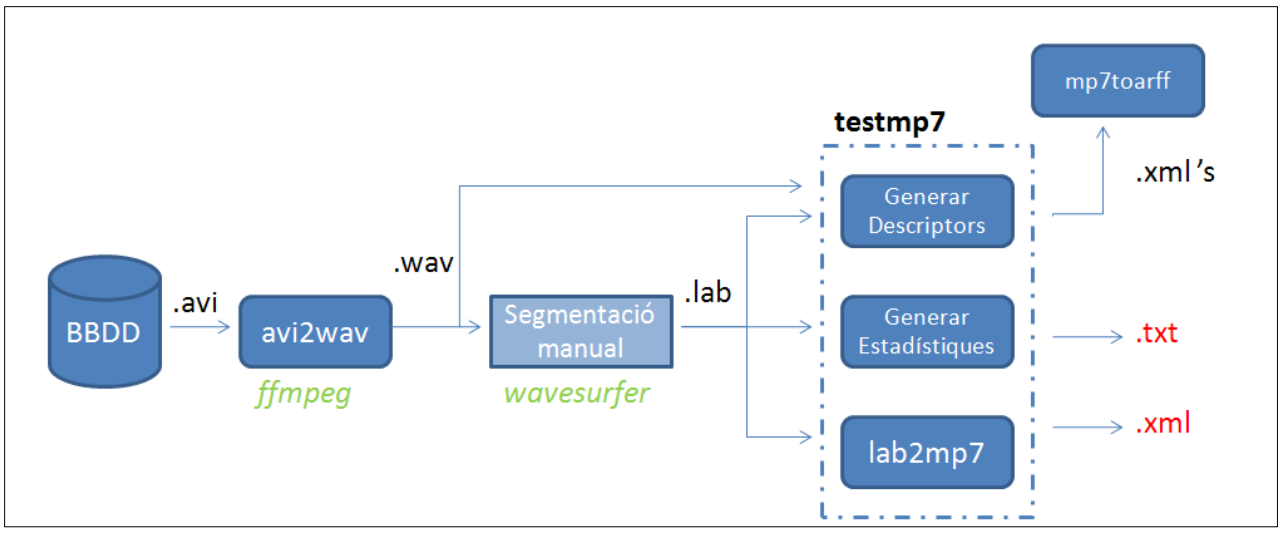


Figura 5.4. Esquema generació de descripció.

Generar descriptors es troba dins del programa testmp7 tal i com s'ha explicat a l'apartat 4 d'aquest document. Aquest algorisme permet crear una sèrie de documents .xml on resideix la informació que s'extreu de l'àudio a partir dels arxius sonors que surten del mòdul Generar Segments. La informació que s'extreu es selecciona mitjançant un fitxer de configuració, el qual podem anomenat també esquema de classificació. Aquest fitxer l'anomenem allOptions.xml i contemplarà en un principi totes les opcions, tal i com es mostra a la figura 5.5.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <Config xmlns="http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd" xmlns:mp7ae="http://mpeg7audioenc.sf.net/mpeg7audioenc.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Module mp7ae:enable="true" xsi:type="HarmonicSpectralVariation" />
  - <Module mp7ae:enable="true" xsi:type="AudioFundamentalFrequency">
      <lolimit>50.0</lolimit>
      <hilimit>12000.0</hilimit>
    </Module>
    <Module mp7ae:enable="true" xsi:type="HarmonicSpectralCentroid" />
  - <Module mp7ae:enable="true" xsi:type="AudioSpectrumBasisProjection">
      <frames>0</frames>
      <numic>8</numic>
    </Module>
  - <Module mp7ae:enable="true" xsi:type="AudioPower">
      <logScale>false</logScale>
    </Module>
    <Module mp7ae:enable="true" xsi:type="HarmonicSpectralSpread" />
    <Module mp7ae:enable="true" xsi:type="TemporalCentroid" />
  - <Module mp7ae:enable="true" xsi:type="AudioSpectrumFlatness">
      <hiEdge>16000.0</hiEdge>
      <loEdge>250.0</loEdge>
    </Module>
    <Module mp7ae:enable="true" xsi:type="AudioHarmonicity" />
  - <Module mp7ae:enable="true" xsi:type="AudioSpectrumEnvelope">
      <resolution>0.25</resolution>
      <hiEdge>16000.0</hiEdge>
      <dbScale>false</dbScale>
      <loEdge>62.5</loEdge>
      <normalize>off</normalize>
    </Module>
    <Module mp7ae:enable="true" xsi:type="SpectralCentroid" />
    <Module mp7ae:enable="true" xsi:type="HarmonicSpectralDeviation" />
    <Module mp7ae:enable="true" xsi:type="AudioSpectrumCentroidSpread" />
  </Config>
```

Figura 5.5. Fitxer configuració schema.xml

D'aquesta manera, ja disposem de tants fitxer xml com segments d'àudio i el següent pas es adequar les dades per tal d'introduir-les al programa Weka, que és el que s'encarregarà de l'entrenament.

Tal i com s'ha introduït a l'apartat 3 d'aquest document, el format d'entrada del programa de mineria de dades és específic i per tant, hem de realitzar un pas de traducció de les dades, el qual anomenem mp7toarff. Aquest modul rep un directori on resideixen un conjunt d'arxius de descripció xml generats al modul anterior, i crea un sol arxiu .arff.

Degut als diferents formats de representació de les dades que obtenim a partir dels descriptors de baix nivell d'mpeg-7, com ha quedat exposat a l'apartat 2, on es presenten els diferents descriptors de baix nivell, hem de realitzar transformacions per tal de poder compactar-ho tot en un mateix fitxer .arff. Així doncs, la solució adoptada davant d'aquest problema ha estat la següent:

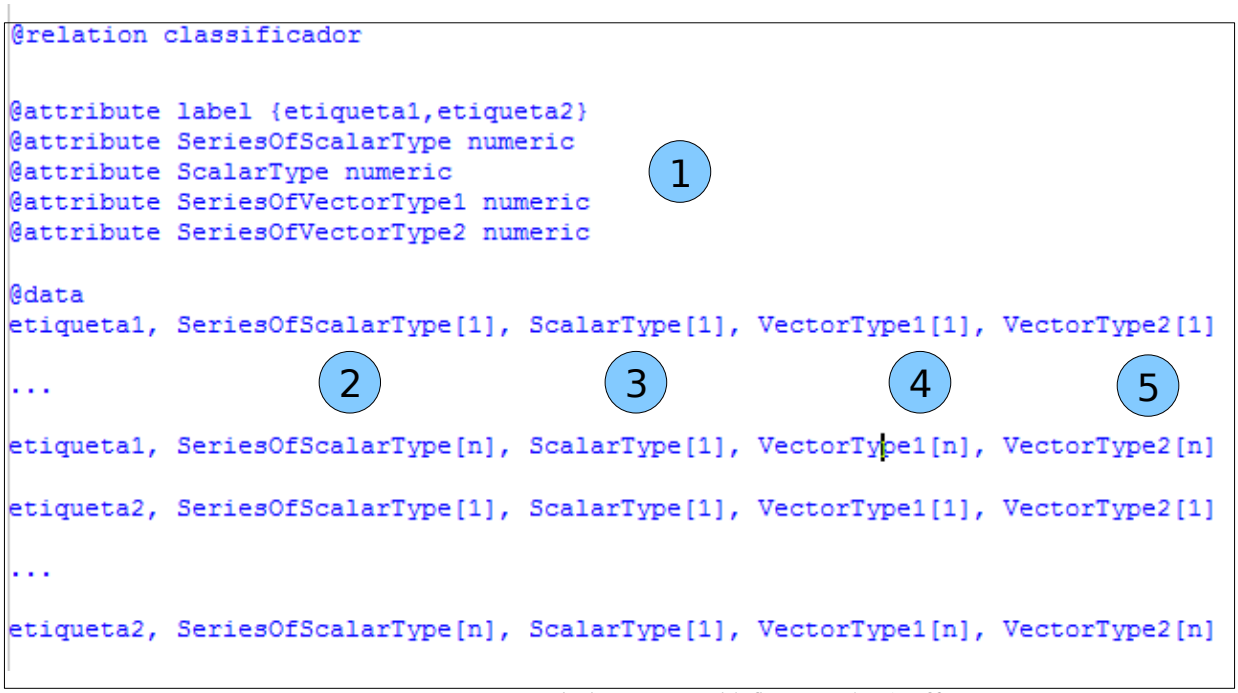


Figura 5.6. Esquema aclariment creació fitxer Weka (.arff)

En (1) es defineixen els atributs, en el nostre cas només utilitzarem els tipus d'arxiu label i numèric. El cas del label correspondrà a les etiquetes assignades a cada segment. El problema el trobem quan es combinen tipus ScalarType, SeriesOfVectorType i SeriesOfScalarType, per la única

raó que el nombre de dades és diferent per a cada cas. ScalarType(3) només té un únic valor, SeriesOfScalarType(2) te N valors, i finalment, SeriesOfVectorType(4),(5) te NxM valors. Així doncs la decisió final ha estat, tal i com es veu a la figura 5.6, repetir el valor ScalarType N vegades per a la mateixa etiqueta, recórrer SeriesOfScalarType i assignar N valors diferents per a la mateixa etiqueta i finalment, per a SeriesOfVectorType definirem M atributs, i recorrerem cada fila M de la mateixa manera que en el cas de SeriesOfScalarType.

D'aquesta manera, ja disposem de les dades adequades per a introduir-les al següent procés, la generació dels models.

5.2 Entrenament i creació dels models

Tal i com ja ha estat introduït dins de l'apartat 1.1, hi ha diferents mètodes establerts pel que fa el procés d'entrenament i classificació de les dades. Els més utilitzats i els que funcionen a l'actualitat són per una banda mètodes estadístics, com ara Hidden Markov Models (HMM) [38], Artificial Neural Networks(ANN)[39], i Nearest Neighbor Rule (NNR)[40] Aquests mètodes s'utilitzen davant d'un ample ventall de problemes, com ara la classificació de música, la identificació de paraules amb un alt percentatge d'encert.

Pel que fa al HMM, es considera una tècnica molt ben establerta en l'àmbit del reconeixement de la parla i el modelat de senyals de veu. ANN, per altre banda, és una tècnica la qual intenta imitar el comportament de aprenentatge humà, utilitzant el concepte de neurona i establint connexions d'entrada i sortida entre elles. En quant a NNR, es tracta d'un mètode bastant senzill, al qual es comparen les característiques dels senyals amb una base de dades entrenada i s'assigna al senyal anònim la classe o etiqueta d'aquesta.

Per una altra banda, existeixen eines de mineria de dades tal i com ha estat presentat a l'apartat 3.1 d'aquests document. Aquestes eines gestionen les dades per tal d'extreure'n informació que resideix en elles. Tot i tractar-les en aquest document de manera independent dels processos esmentats anteriorment, les bases de la mineria de dades es trobem dins de la intel·ligència artificial i l'anàlisi estadístic. D'aquesta manera, mitjançant la mineria de dades es creen models que permeten realitzar tasques de predicció, classificació i segmentació.

Així doncs, és el moment d'endinsar-nos dins del programa Weka. Tal i com s'ha presentat a l'apartat 3, existeixen molts mètodes classificadors, alguns dels quals s'han esmentat amb anterioritat.

L'elecció del classificador a utilitzat no és trivial, si no que es basa en diferents conclusions extretes a partir de realitzar diferents proves i experiments, les quals l'eficiència del classificador i la facilitat d'extreure el model són les característiques que més s'han valorat. Així doncs, després de realitzar proves amb els classificadors J48Tree, RandomTree, NNR i NBTree, es va seleccionar el mètode RandomTree.

Com el mateix nom indica, el classificador RandomTree genera un arbre classificador on es relacionen els diferents atributs. Va ser el classificador escollit degut a l'alt percentatge de classificacions correctes, la facilitat d'accedir al model mitjançant l'arbre que posteriorment es podrà implementar a partir de sentències condicionals, i els bons resultats que presenta el programa en quant a eficiència. A més, aquest classificador permet seleccionar la profunditat de l'arbre de classificació, la qual cosa implica la possibilitat de reduir al màxim les decisions i establir un llindar únic per la classificació.

Primer experiment de classificació "tots"

Així doncs, com a primer pas es crea el primer arxiu d'entrada a weka corresponent a tots els arxius de descripció (xml) corresponents als tipus SeriesOfScalarType i ScalarType, és a dir, amb les etiquetes establertes anteriorment: home, dona, silenci, soroll, música.

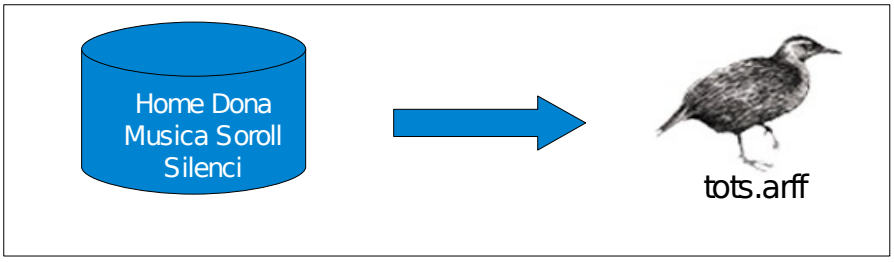


Figura 5.7. Generació fitxer Weka per l'experiment "tots"

Per tal de crear aquest arxiu, primer es visualitzen de manera manual alguns fitxers de descripció de manera aleatòria. El resultat d'aquest visualització és correcte i es veu que el paràmetre Fundamental Frequency es troba repetit dues vegades, essent així una sèrie d'escalars de longitud doble en comparació amb tots els altres descriptors del tipus SeriesOfScalar. Així doncs, es considera oportú delmar aquest descriptor de manera que quedi només un valor dels repetits dues vegades, i d'aquesta manera, del mateix longitud que tots els descriptors del tipus SeriesOfScalar.

El resultat d'aquest arxiu Weka és un fitxer on es descriuen les característiques dels diferents sons corresponent a les etiquetes donades manualment. En aquesta primera fase, les característiques seleccionades són totes. En conseqüència, l'arxiu resultat ha estat de més de 100MB. Això ha comportat dos problemes. Per una banda, el programa no és capaç de carregar tal quantitat de dades. La primera solució que s'adopta és modificar el fitxer de configuració del programa RunWeka i augmentar la memòria RAM que té assignada. Així doncs es procedeix canviant el paràmetre *maxheap* que inicialment estava a 128m, per 512m.

Un cop canviat aquest paràmetre el programa obre l'arxiu de descripció arff, però alhora de treballar dins del programa, s'excedeix aquesta memòria RAM. Aquest es el segon problema, que es resol delmant les dades per un factor de 5 en tots els casos i de 10 en el cas de Fundamental Frequency. És ara quan el programa pot treballar bé i es poden realitzar les primeres proves. Es defineixen els següents paràmetres per la classificació:

- Classifyy method:RandomTree
- Max Depth of the Tree: 8
- Test Mode: split 70% train, remainder test
- Attributes: label, HarmonicSpectralCentroid, HarmonicSpectralDeviation, HarmonicSpectralSpread, HarmonicSpectralVariation, AudioFundamentalFrequency, AudioSpectrumSpread, AudioSpectrumCentroid, AudioPower.

El resultat de la classificació és un arbre de 247 branques amb una profunditat de 8. Un arbre d'aquestes característiques és massa gran i en conseqüència, tendeix a individualitzar els camins. Així doncs aquest resultat no és bo per elaborar un classificador. Pel que fa als resultats obtinguts, en general són força bons, però era d'esperar ja que aquest és un cas de sobre-entrenament que no serveix per construir un classificador generalitzat.

```
Correctly Classified Instances       65512               94.4753 %
Incorrectly Classified Instances      3831                5.5247 %
Kappa statistic                          0.9072
Mean absolute error                      0.0358
Root mean squared error                  0.1347
Relative absolute error                 14.7747 %
Root relative squared error             38.7173 %
Total Number of Instances            69343

        === Confusion Matrix ===

           a     b     c     d     e   <-- classified as
       27325   202   518     0    11 |     a = home
        1340  6095    53     0    44 |     b = dona
         802     5 31662     0     6 |     c = musica
           0     0    21   122     0 |     d = silenci
         667    54   108     0   308 |     e = soroll
```

Figura: 5.8 Matriu de confusió i resultats corresponents a l'experiment "tot"

A continuació es realitzen altres proves modificant el paràmetre de profunditat d'arbre. Els resultats van empitjorant a mesura que es disminueix el valor. Arribant a no classificar algunes de les etiquetes. En conseqüència, es decideix estudiar la classificació de manera més específica per a casa problema. Així doncs es defineixen tres passos, el primer de tots, és diferenciar si l'àudio correspon a un segment on hi ha silenci, o per contra, hi ha variació de pressió. Si estem davant un segment on no hi ha silenci, s'estudiarà si aquesta contribució correspon a música o a veu. Un cop aquesta distinció es passarà a reconèixer de quin tipus de veu es tracta, el primer pas, doncs, serà distingir entre veu masculina i femenina. Un cop arribat a aquest punt, es podrà estudiar distingir entre diferents locutors del mateix gènere.

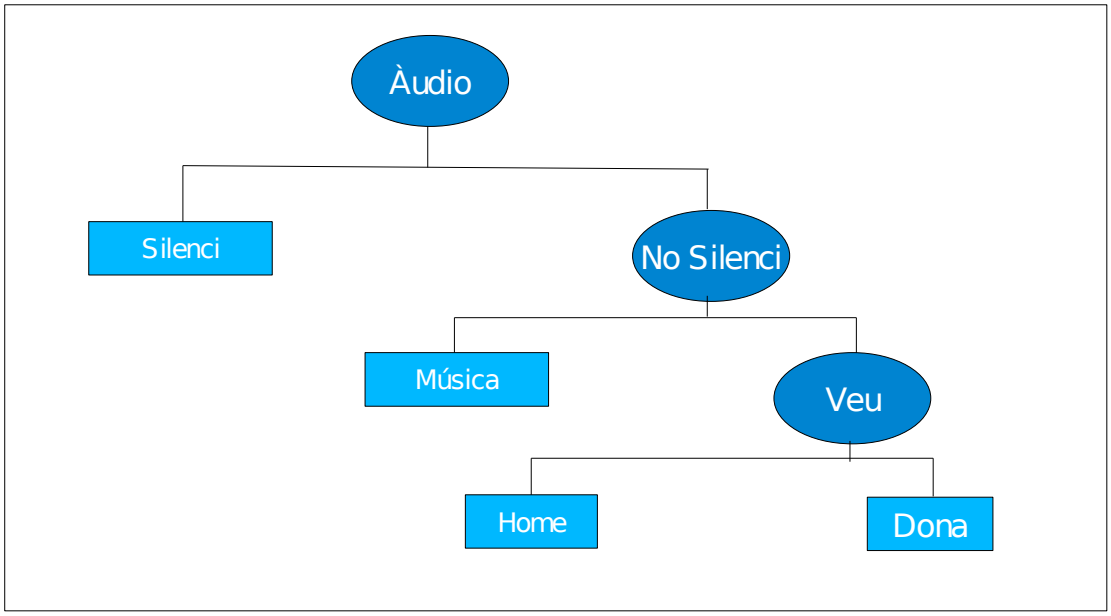


Figura 5.9. Arbre general de classificació basat en relacions entre les etiquetes

5.2.1 Discriminació d'esdeveniments sonors

En aquest punt s'explicaran tots els estudis que s'han anat fent fins a aconseguir els models de classificació final, tant de silenci-no silenci, com de veu-música.

5.2.1.2 Discriminació silenci-no silenci

El primer pas per tal de determinar el contingut de l'àudio passa per determinar si en un punt trobem o no silenci. Existeixen múltiples formes de fer-ho però s'ha determinat provar una idea que a priori sembla lògica.

Com a idea principal, teòricament bastaria amb el descriptor d'AudioPower (AP) per a poder discriminar entre ambdues classes. Bastaria fixar un llindar per sota del qual avaluem la mostra com soroll. Una primera part d'aquest punt seria fer l'estudi que pugui determinar aquest llindar. L'altre punt consisteix en determinar el temps que l'àudio s'ha de mantenir sota aquest llindar per a que es pugi considerar que es un moment de silenci.

La part corresponent al temps mínim per a etiquetar un segment com a silenci ja ha sigut resolta , ja que quan es va dissenyar l'arbre d'etiquetat i es van fixar les normes d'etiquetat es va fixar el temps mínim de silenci con 2 segons. Aquest paràmetre s'ha fixat d'acord amb les voluntats que té VSN respecte a l'ús d'aquest classificador. Per a determinar el llindar, s'ha fet un estudi dels

valors de AP de tots els segments de veu i música utilitzats en l'entrenament. Per fer aquest estudi, s'ha modificat l'algoritme MPEG-7toarff per tal que generi un arxiu amb els valors d'AP de cada segment. Aquest descriptor es del que retorna una serie d'escalars, el que significa que tindre un valor d'AP per cada 10 ms. d'àudio. Els resultats es poden veure gràficament en la següent figura:

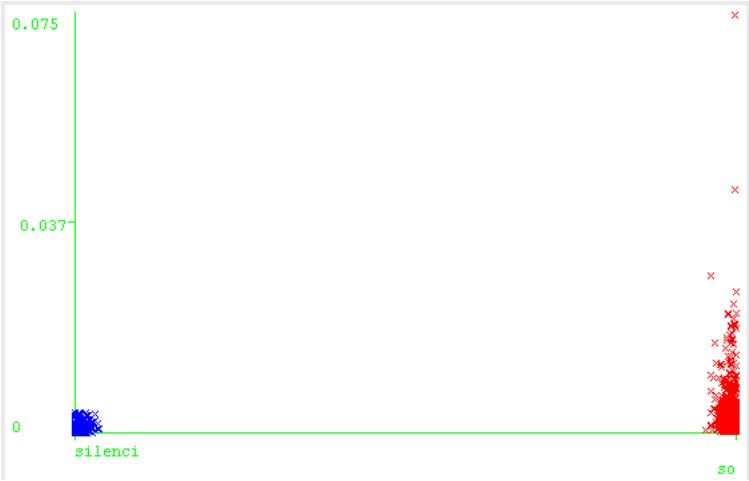


Figura 5.10. Gràfica dels valor d'AP del segments de silenci i só.

Com es pot veure, els valors de silenci estan molt a prop de zero. Tot i això, hi ha parts de veu que les classifica com a silenci. El poc encert del classificador es deu en gran part a que el valors introduïts d'AP són cada 10 ms. De forma que hi ha moments de molt poca durada dins de segments que no són silencis que potser tindran un valor d'AP molt baix i que donen peu a aquests resultats tan pobres.

Després de fer un estudi manual de les mostres d'àudio , s'ha pogut comprovar que el segment amb un valor mig més baix es troba amb una magnitud de 1xE-5. Per tant, s'ha situat el llindar a 8xE-6. La forma de fer aquests estudi ha sigut fent la mitjana dels valors d'AP de cada segment de so i després buscar el valor mínim de la seqüència.

Així doncs, s'ha dissenyat un arbre classificador com el que es mostra a la figura 5.11, on es valora cada mostra d'AP i quan passen més de dos segons (200 mostres ja que es pren una mostra cada 10 ms.)per sota del llindar, es comença a contar com a silenci.

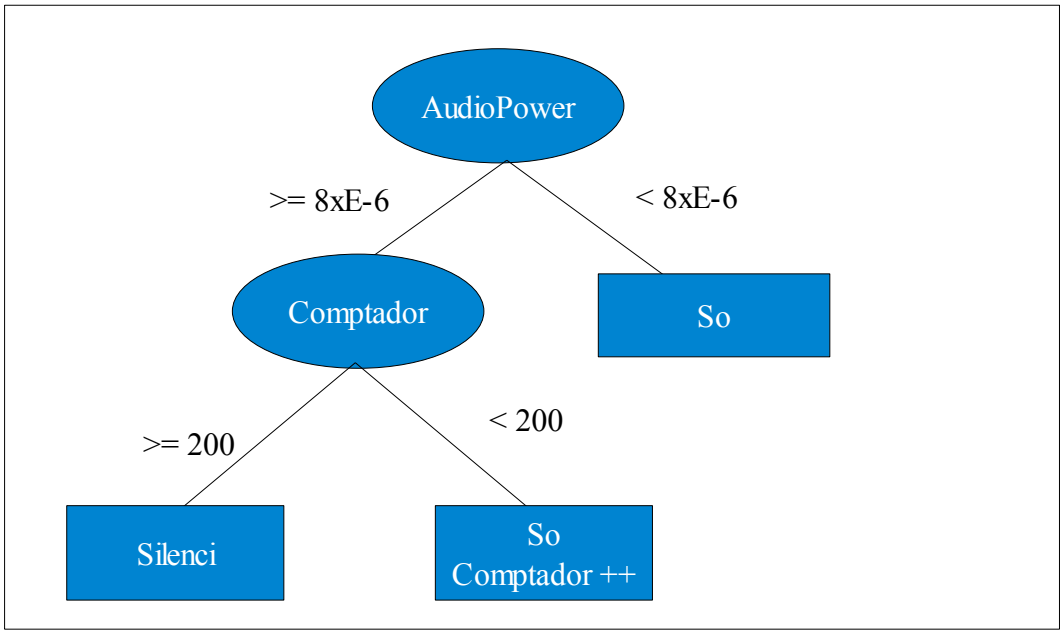


Figura 5.11 Arbre classificador de Silenci-no silenci.

5.2.1.2 Discriminació veu-música

Per tal d'obtenir el model del classificador veu-música , s'ha portat a terme un estudi per tal de trobar quins descriptors son més determinants a l'hora de separar música i veu.

El primer pas ha sigut d'entre tota la base de dades de segments d'àudio, crear una carpeta que només tingui aquests dos tipus d'arxius, veu i musica. Per això s'ha modificat el programa "testMPEG-7" de forma que .només crea els MPEG-7 d'aquestes classes d'arxius. Després, s'ha modificat també el programa "MPEG-7toarff" per tal d'obtenir un arxiu arff que contingui només els labels: veu i musica. Al final hem queda una carpeta " veu_i_musica" amb tots els arxius de veu pura i musica.

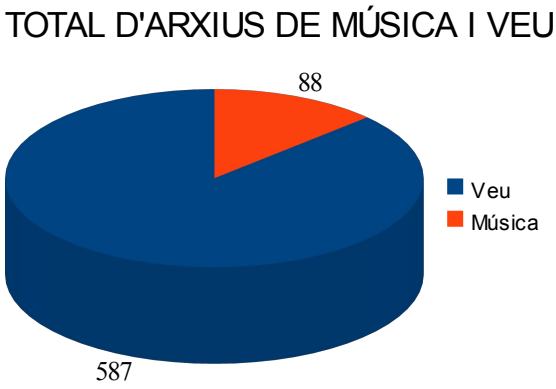


Figura 5.12. Representació del contingut de la carpeta Veu_i_musica abans de la millora

En aquesta gràfica sobre el contingut de la carpeta, podem observar que la proporció de veu respecta a música es molt alta. Aquesta gran diferencia pot portar conseqüències negatives a l'hora d'entrenar el classificador, ja que generarà el model a partir de molt poques mostres de música en relació amb la veu, així que el primer pas ha sigut depurar la base de dades incrementant els segments de música.

MILLORA DEL CONTINGUT DE LA BASE DE DADES

En un intent per millorar el resultat de les probes fetes, s'ha determinat augmentar el nombre d'arxius de musica que utilitzo per entrenar el sistema. Per això, 166 arxius wav de musica de la pagina del ministeri d'educació de l'esta han sigut descarregats. Aquests arxius corresponen a músiques d'estils i
duracions molt variats. Un cop fets els MPEG-7,han sigut ajuntats a la base de dades . Per altra banda, s'han eliminat un petit numero d'arxius de veu que no eren prou "purs" i que podien distorsionar malmetien les gràfiques i els resultats.

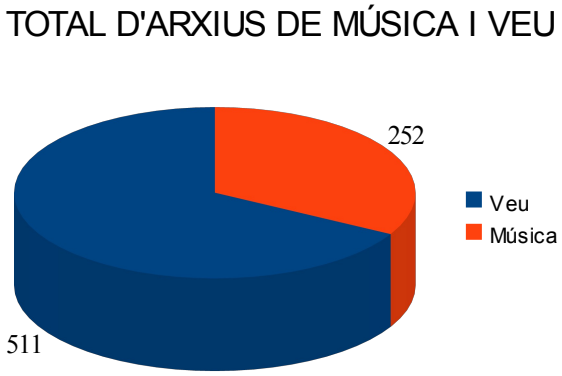


Figura 5.13 Representació del contingut de la carpeta Veu_i_musica després de la millora

ESTUDI D'UN DESCRIPTOR COM A POSIBLE FACTOR DETERMINANT

Un cop la base de dades ja era més fiable, s'ha començat l'estudi dels diferents descriptors.

En primera instància s'ha estudiat l'Audio Spectral Spread(ASS). Aquest descriptor ens determina si el contingut espectral es troba al voltant del centroide o pel contrari, es troba repartit per tot l'espectre.

Tal com s'indica [4] , les veus haurien de tindre un valor de ASS més baix, ja que en teoria l'espectre no hauria de desplegar-se per tot el rang freqüencial. No obstant, l'anàlisi no mostra el mateix.

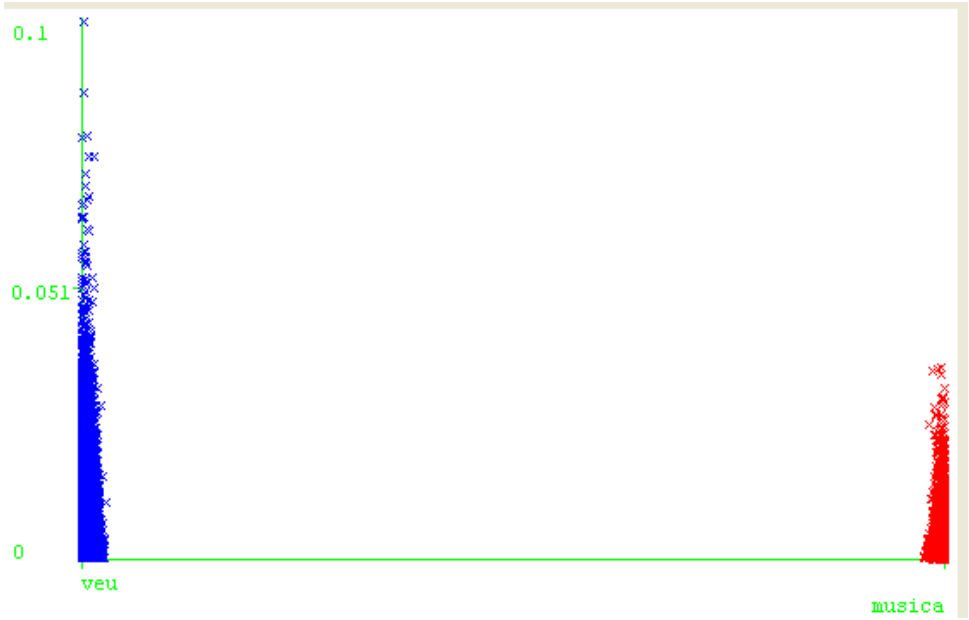


Figura 5.14 Gràfica dels valors d'ASS de totes les mostres de veu i musica

A la figura 5.11 es mostren els valors recollits de totes les mostres de veu i àudio. Com es pot observar, no trobem una separació clara entre els valors de cada classe. Ambdues tenen el gruix dels seus valors a la mateixa zona.

A continuació, i per tal de verificar aquests resultats . s'ha fet una avaluació de dos arxius concrets (naturalment10_1_musica1.wav i setdies_178_home1.wav). Ambdós tenen una durada de 29 segons i corresponen a un arxiu de veu d'home i un arxiu de música. Els dos han estat escollits tenint en compte que fossin el més purs possibles. L'anàlisi dels valors de ASS que presenten els seus MPEG-7 es el següent (Nota: els valors han sigut multiplicats per 100 per tal de que es pugui veure més clara la gràfica i a més se'ls ha aplicat un filtrar pas baix degut a les grans oscil·lacions que presentaven)

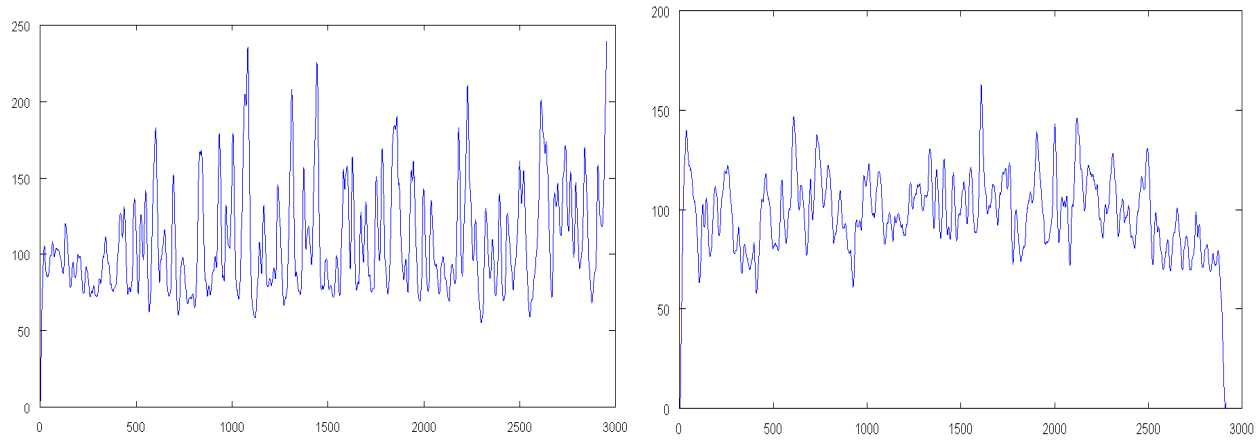


Figura 5.15. Esquerra: Valors d'ASS del segment de veu. Dreta: Valors d'ASS del segment de música

Es pot observar que en els dos casos els valors es troben al voltant de 1. Tot i que sembla que en general la gràfica de música es troba més amunt, els pics que presenta la gràfica de veu fan que el resultat sigui molt semblant. A la figura 5.13 es veu més clarament en els histogrames de les dades filtrades:

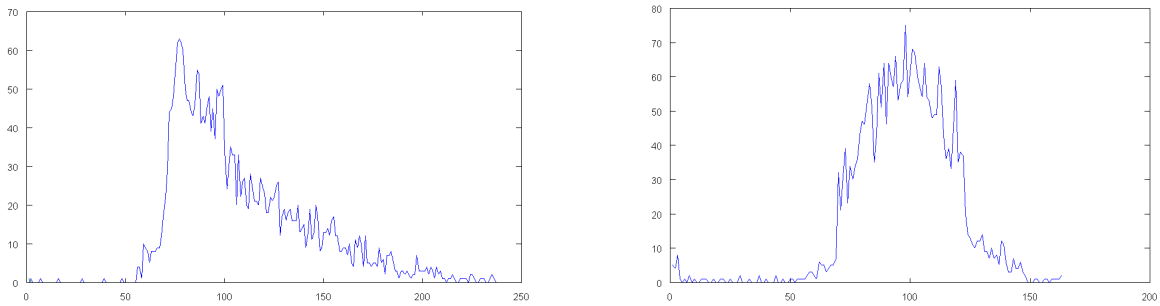


Figura 5.16 Esquerra: Histograma de ASS del segment de veu. Dreta: Ídem per segment de música

Aquí es veu com sembla que la veu estaria més avall, però poc (sobre 80 la veu i sobre 100 la música). Per tant, en arxius que fossin menys purs, encara seria més difícil de diferenciar. Es per això que s'ha descartat l'us, almenys únic, d'aquest LLD.

ESTUDI DELS ALTRES LLD PER SEPARAT

Un com estudiat el descriptor que semblava que seria més significatiu, s'ha procedit a fer un estudi de la resta de descriptors que sembla que poden ser importants. De l'estudi fet sobre un altre classificador d'àudio [12], s'extreu un llista d'onze LLD d'MPEG-7 que són útils per a classificar. D'aquests onze, però, s'ha escollit utilitzar nou, deixant de banda els que retornen un vector de dades, degut a la enorme quantitat d'informació que generen.

Descriptor	Dimensió
Audio Spectrum Centroid (ASC)	Series of Scalar
Audio Spectrum Spread (ASS)	Series of Scalar
Audio Fundamental Freq (AFF)	Series of Scalar
Harmonic Ratio (HR)	Series of Scalar
Spectral Centroid (SC)	Scalar
Harmonic Spectral Centroid (HSC)	Scalar
Harmonic Spectral Deviation (HSD)	Scalar
Harmonic Spectral Spread (HSS)	Scalar
Harmonic Spectral Variation (HSV)	Scalar

Figura 5.17. Taula del LLD utilitzats en l'estudi i el tipus que retornen

L'estudi del ASS ja s'ha portat a terme, per tant a continuació s'ha fet el mateix estudi amb la resta dels descriptors, utilitzant els mateixos dos arxius. Aquí s'ha de tindre en compte que els descriptors que retornen un únic valor no se'n pot fer l'histograma, per tant es fan dels que retornen una serie, i els altres mirarem els valor que extreuen.

AudioFundamentalFrequency

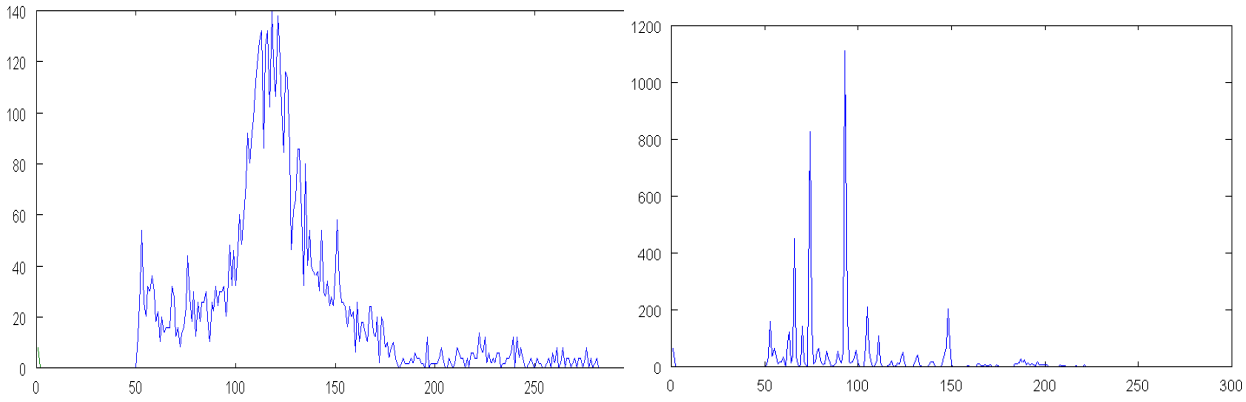


Figura 5.18. Esquerra: Valors d'AFF del segment de veu. Dreta: Valors d'AFF del segment de música

Tot i que les veus presenten un major rang de valors en front de valors contrets de la música, la gran majoria semblen estar centrades sobre els mateixos valors. No obstant, la música presenta pics a freqüències inferiors a les de la veu, fet que podria determinar una frontera entre veu i música, potser per si sola no totalment determinant però que podria ajudar barrejada amb altres descriptors.

HarmonicRatio

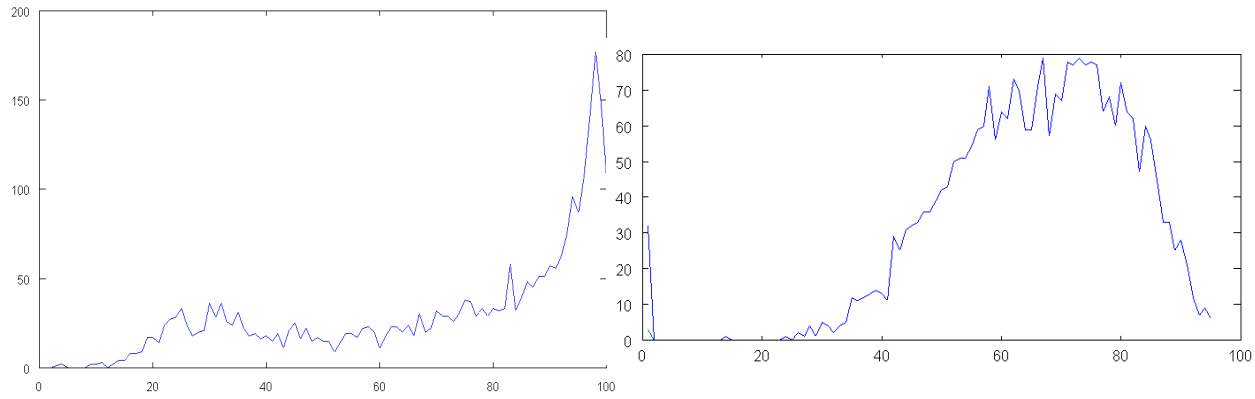


Figura 5.19 Esquerra: Valors d' HR del segment de veu. Dreta: Valors d' HR del segment de música

Els valors d'HR presenten una incongruència. Segons la teoria, una peça musica presenta valors superiors als que presenta una veu. Les causes poden ser múltiples: mala implementació del descriptor, mala elecció de les mostres, una simple anomalia...

AudioSpectrumCentroidType

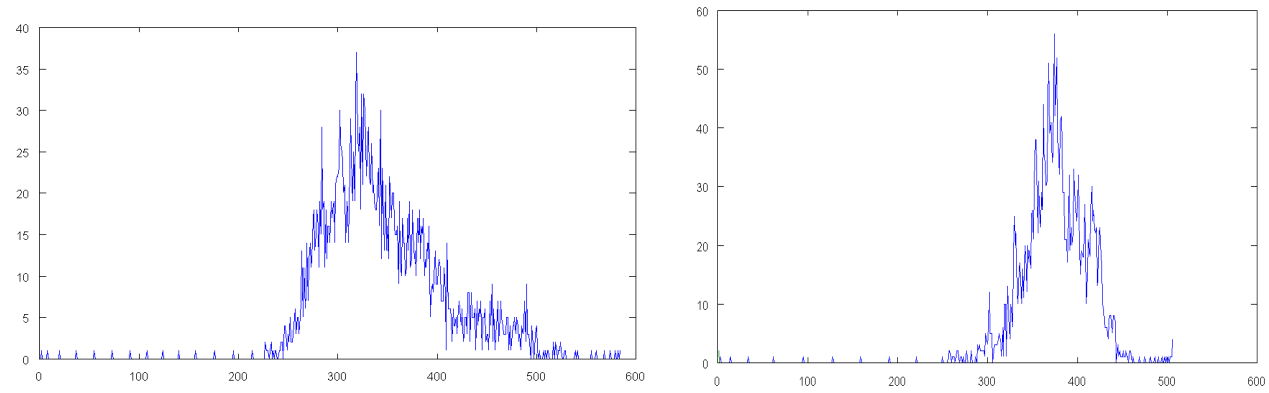


Figura 5.20 Esquerra: Valors d' ASC del segment de veu. Dreta: Valors d' ASC del segment de música

Aquests histogrames si que mostren una separació clara entre ells. A més, els valors obtinguts són lògics i corresponen a l'esperat. Aquest es el descriptor que a priori sembla que pot donar resultats. No obstant, tampoc sembla ser decisiu. Tot i que la música presenta pics per sobre dels de la veu i a més es trobe dins d'un marge de valors reduït, hi ha masses valors de la veu que es troben dins del marge on actua la música.

A continuació, ja que no es pot fer un histograma d'un sol valor, es mostren els valors obtinguts dels descriptors que retornen un escalar a la figura 5.21.

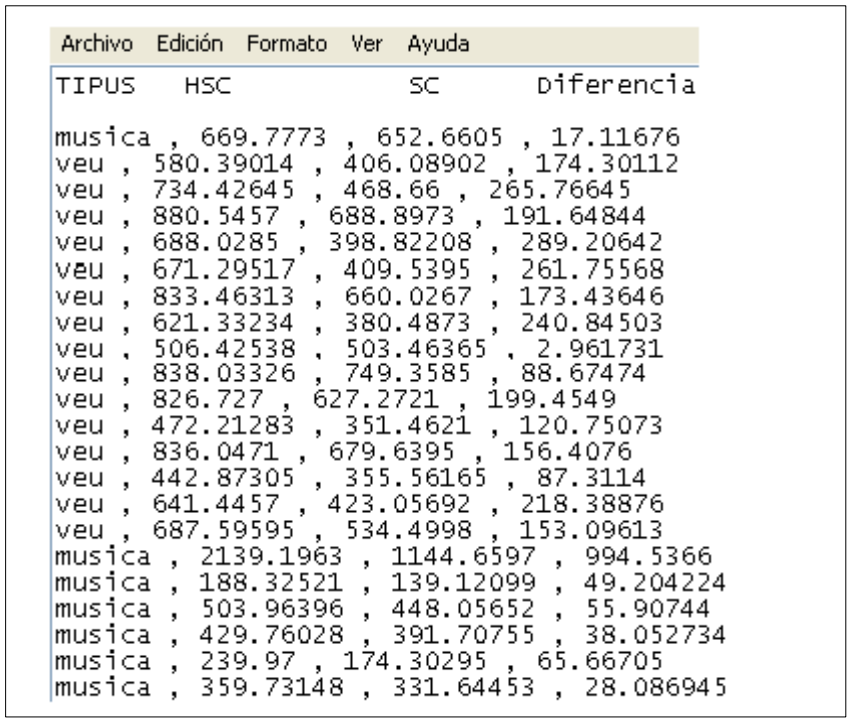
	SpectralCentroid	HarmonicSpectral Centroid	HarmonicSpectral Deviation	HarmonicSpectral Spread	HarmonicSpectral Variation
Veu home	452.47284	699.786	0.17612584	0.54955167	0.17513557
Musica	562.734	564.77856	0.20134994	0.49059713	0.23540011

Figura 5.21 Taula de valors dels diferents descriptors per als arxius de veu i música analitzats.

Només amb aquests descriptors no sembla que es pugui extreure molta informació. No obstant, sembla que el Spectral Centroid (SP) i el Harmonic Spectral Centroid (HSP) de la música són molt propers (dada lògica i a esperar), dit d'una altra manera, hi ha correlació entre ells, mentre que en la veu d'home no. Per tal de veure si això es cert, s'ha fet un estudi sobre el tema.

ESTUDI DE LA CORRELACIÓ ENTRE SP I HSP

Per tal de veure si la teoria esmentada anteriorment té base sòlida, primer s'ha modificat el programa mp7toarff per a que generi un arxiu .txt on mostri el valor de SP HSP i l'absolut de la diferencia d'aquests. D'aquests arxiu es pot veure, com reflexa la figura X que encara que no és en un cent per cent dels cassos, si que és més comú que els segments de música presentin valors diferencia més baixos que els segments de veu.



Archivo Edición Formato Ver Ayuda

```
TIPUS    HSC            SC       Diferencia

musica , 669.7773 , 652.6605 , 17.11676
veu , 580.39014 , 406.08902 , 174.30112
veu , 734.42645 , 468.66 , 265.76645
veu , 880.5457 , 688.8973 , 191.64844
veu , 688.0285 , 398.82208 , 289.20642
veu , 671.29517 , 409.5395 , 261.75568
veu , 833.46313 , 660.0267 , 173.43646
veu , 621.33234 , 380.4873 , 240.84503
veu , 506.42538 , 503.46365 , 2.961731
veu , 838.03326 , 749.3585 , 88.67474
veu , 826.727 , 627.2721 , 199.4549
veu , 472.21283 , 351.4621 , 120.75073
veu , 836.0471 , 679.6395 , 156.4076
veu , 442.87305 , 355.56165 , 87.3114
veu , 641.4457 , 423.05692 , 218.38876
veu , 687.59595 , 534.4998 , 153.09613
musica , 2139.1963 , 1144.6597 , 994.5366
musica , 188.32521 , 139.12099 , 49.204224
musica , 503.96396 , 448.05652 , 55.90744
musica , 429.76028 , 391.70755 , 38.052734
musica , 239.97 , 174.30295 , 65.66705
musica , 359.73148 , 331.64453 , 28.086945
```

Figura 5.22 Exemple de l'arxiu de text creat per tal d'avaluar la hipòtesi.

Per tant, el següent pas ha sigut crear un atribut que passar-li a weka anomenat diferencia, per tal de veure quins percentatges de classificació s'aconsegueixen utilitzant aquest valor.

CREACIÓ DE CLASIFICADOR D'UN SOL DESCRIPTOR

Ja que d'entrada ha sigut impossible trobar un descriptor que pugi discretitzar fàcilment entre música i veu, amb l'eina de data mining weka s'ha avaluat si pot crear un arbre que permeti discretitzar només utilitzant un descriptor, i quins percentatges d'encert podria donar..

Per tal de poder crear el model l'algoritme de classificació emprat és el exposat en el punt 5.1, el random tree, amb els arxius de la base de dades utilitzats per a l'entrenament tal i com s'exposa en el mateix punt. Ja que el programa de data mining, tal com s'explica en el capítol 3, ja fa proves amb el model classificador que genera, podrem veure a partir del seu percentatge d'encert quin descriptor sembla donar millors resultats. A la figura 5.X podem observar els valors resultants al descriptors que retornen una serie d'escalars.

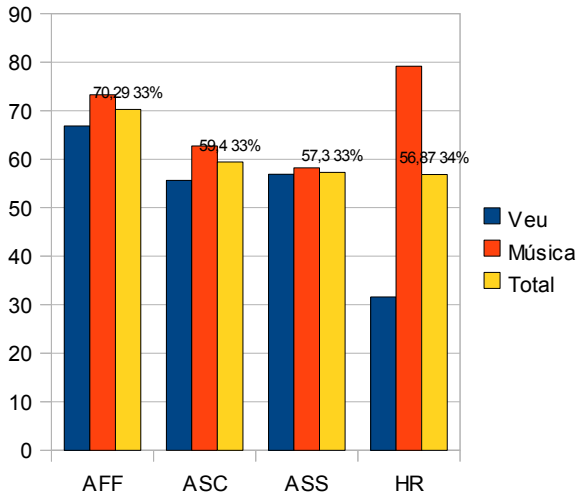


Figura 5.23 Gràfica amb els percentatges d'encert dels descriptors que retornen un SeriesOfScalar

Els resultats venen a confirmar en part les conclusions obtingudes anteriorment, encara que a més afegeixen informació al respecte dels descriptors. És estrany, ja s'ha dit abans, que el descriptor ASS presenti un percentatge d'encert tant baix, sobre quasi el més baix de tots. En canvi, com s'havia la freqüència fonamental sembla donar, situant la frontera sobre els 100 Hz, prou bons resultats. Una altra dada que es pot extreure i que ja abans s'havia pogut entreveure i ara ha quedat confirmada és el fet de la bona classificació que fa l' Harmonic Ratio de la música, la millor de tots, que es veu clarament disminuïda per la mala que fa de la veu, la pitjor de totes quatre. Aquesta diferencia es deu al fet que la música presenta uns valors HR definits dins d'un marge, en canvi la veu, possiblement degut a moments puntuals, presenta un marge molt més gran on molt valors cauen dins el rang de la música.

A continuació es procedeix a fer el mateix estudi amb els descriptors que retornen un sol valor.

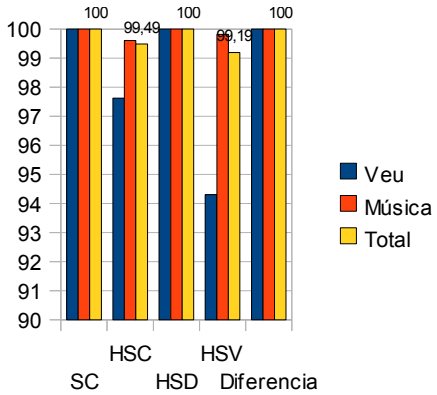


Figura 5.24 Gràfica amb els percentatges d'encert dels descriptors que retornen un Scalar

Els resultats, excel·lents a simple vista però molt dubtosos veient els resultats obtinguts anteriorment, tenen un explicació. El fet de tenir aquests percentatges tant elevats es deu al fenomen del sobre-entrenament.. Tal com s'ha pogut comprovar, aquest fenomen succeïx degut a la forma com s'ha dissenyat l'arxiu de dades d'entrada al programa. El fet de repetir un valor únic de cada mostra un nombre de vegades igual a la seva duració multiplicat per mil, dona molts valors idèntics. Això fa que el programa identifiqui aquest punts concrets i crei un arbre especialitzat en aquests valors .

Per a corregir-ho, s'ha re dissenyat el codi que genera el arxiu arff de tal forma que crea dos arxius en comptes d'un sol. Per una banda, genera un arxiu amb els descriptors que retornen un SerieOfScalar i per l'altra un amb els que retornen un Scalar. Ara els valors no estan repetits, i quan és tornen a fer les probes els resultats són els següents:

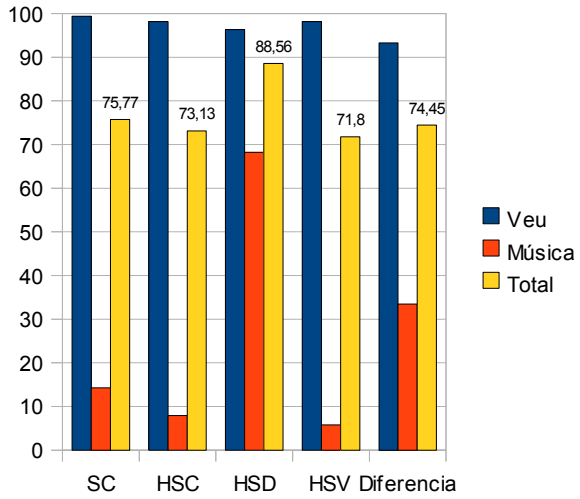


Figura 5.25 Gràfica amb els percentatges d'encert corregits dels descriptors que retornen un Scalar

Com es pot observar, l'error ha sigut corregit i ara ja es pot veure com classifiquen aquests descriptors. A simple vista, destaca la gran diferencia a l'hora de classificar cada etiqueta. Mentre que amb les veus la classificació és optima, el percentatge amb la música es pèssim. Només el HSD obté bons resultats, i es postula com a descriptor a utilitzar. Es pot veure que el valor diferencia també aconsegueix millors resultats en aquest aspecte, tot i no ser tant bons com els esperats.

De totes maneres, el fet que el contingut de veu sigui major que el música, fa que el resultat final d'aquests classificadors sigui molt millor, de l'ordre d'un vint per cent , als dels altres descriptors. Això sumat al fet que dins els arxius per als quals vol encarar-se aquest projecte contenen majoritàriament veu, es determina com a primer experiment dissenyar el classificador a partir d'aquests descriptors.

DISSENY FINAL DEL CLASSIFICADOR

A partir dels estudis fets anteriorment s'ha arribat a una conclusió. Per tal de no barrejar tipus de descriptors i facilitar el disseny, i degut als resultants obtinguts, s'utilitzaran els descriptors que retornen un escalar com a base per a dissenyar el classificador. D'aquests, s'han utilitzat aquells dos que millors resultats han donat, es a dir, HSD i Diferencia,tot i que es contempla la possibilitat d'incorporar un tercer. El que fem llavors es introduir al weka els descriptors que volem utilitzar, limitar les branques de l'arbre de sortida a un màxim de dues per descriptor, i comprovar els resultats. El classificador que millors resultats ha donat ha sigut el següent:

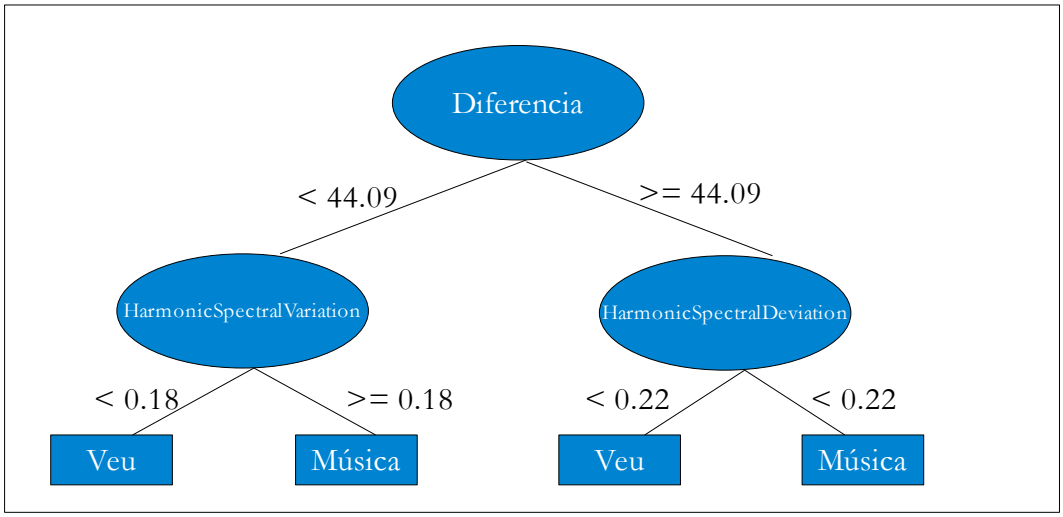


Figura 5.26 Arbre classificador de Veu-Música

5.2.2. Discriminador de gènere del locutor

Un cop disposem de la sortida del discriminador de diferents esdeveniments sonors, disposem d'una base de dades segmentada en tres tipus: música, veu i silenci. És ara quan ens podem valdre de tota la base de dades corresponent a veu i estudiar les característiques de la veu.

Una de les característiques més importants pel que fa a la veu, és distingir si el locutor és masculí o femení. Així doncs, el primer pas serà estudiar les característiques de l'àudio que diferencien una veu femenina d'una masculina.

Per tal d'introduir les dades a Weka, creem un arxiu .arff on només estudiem les característiques de la base de dades destinada a training corresponent a les veus. El directori on resideixen tots els fitxers de descripció es mesclen arxius dels tres programes utilitzats pel procés d'entrenament: banda1, banda2, naturalment8. El nostre algorisme mp7toarff permet crear l'arxiu arff específic per a aquest experiment.

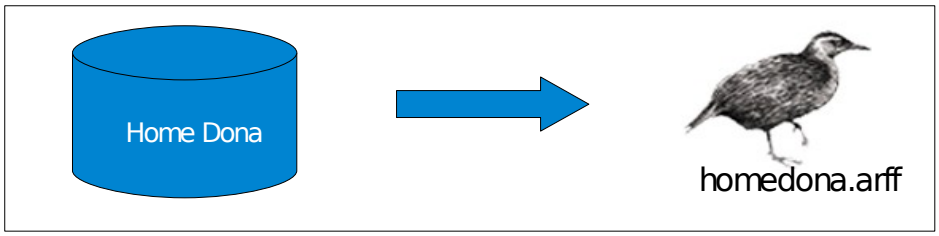


Figura 5.27. Creació arxiu homedona.arff

En una primera fase de l'experiment, seleccionem tots els descriptors corresponents als tipus SeriesOfScalarType i ScalarType, i iniciem el procés de classificació. Per tal de realitzar la classificació es selecciona el mètode classificador RandomTree, ja que com s'ha explicat anteriorment, és el que permet retallar la profunditat de l'arbre.

```
@attribute label {home,dona}
@attribute HarmonicSpectralCentroid numeric
@attribute HarmonicSpectralDeviation numeric
@attribute HarmonicSpectralSpread numeric
@attribute HarmonicSpectralVariation numeric
@attribute AudioFundamentalFrequency numeric
@attribute AudioSpectrumSpread numeric
@attribute AudioSpectrumCentroid numeric
@attribute AudioPower numeric
```

Figura 5.28. Atributs seleccionats per homedona.arff

En aquest cas, ja que son vuit els atributs seleccionats, es modificarà el paràmetre *maxDepth* per que sigui vuit. Els resultats és un arbre de classificació de 149 branques i els resultats de la classificació són els següents:

```
Correctly Classified Instances       34996               97.7952 %
Incorrectly Classified Instances       789                2.2048 %
                === Confusion Matrix ===

     a     b   <-- classified as
 27877   349 |     a = home
   440  7119 |     b = dona
```

Figura 5.29 Resultats classificació per homedona.arff

Els resultats de classificació són molt bons, però, tot i haver-hi limitat la profunditat de l'arbre, aquest és massa gran, i en conseqüència massa selectiu. Això porta a la hipòtesi d'obtenir un classificador sobre-entrenat, és a dir, molt específic per a les dades seleccionades d'entrada. Per tant, no són uns models vàlids, ja que l'objectiu és elaborar un classificador el més genèric possible.

Així doncs, per tal de conèixer quins son els descriptors més significatius per la discriminació entre home i dona, s'estudiaran els descriptors de manera individual a fi de determinar possibles tractaments d'aquests que millorin la classificació.

ESTUDI DEL DESCRIPTOR "AUDIO FUNDAMENTAL FREQUENCY"

La característica més diferenciadora entre una veu masculina i una veu femenina, a simple vista i per a qualsevol persona, és el to. Així doncs, es diu que una veu masculina és greu, i una femenina és aguda. En conseqüència, el primer descriptor a estudiar, és el que relaciona el to dels sons, la freqüència fonamental. La freqüència fonamental és la freqüència més baixa a la qual un so periòdic comença a vibrar. Aquesta es troba repetida al llarg de tot l'espectre com a múltiples de la fonamental, i s'anomenen harmònics. En el cas de la veu humana, s'han realitzat estudis[41] on es demostra que la freqüència fonamental pot anar des de 100Hz- 200Hz en el cas dels homes i de 150-300 en el cas de les dones, essent la veu dels infants la més aguda superant els límits assignats a les veus femenines.

Per veure quin és el pes que fa a la classificació aquest paràmetre, es carregarà a Weka l'arxiu creat anteriorment i a la pestanya de pre-processat només es seleccionarà els atributs *label{home,dona],* i *AudioFundamentalFrequency.* Es realitzarà la prova amb un 70% de dades per entrenar i la resta per test i amb una profunditat màxima de 1.S'observa que els valors que s'obtenen en alguns casos passen el valor d'1KHz. Segons els valors donats anteriorment pel que fa als límits de la freqüència fonamental, això no és possible. Així doncs, hi ha un error a l'hora d'extreure les característiques de l'àudio.

Per tant, abans de realitzar més proves es modifica l'esquema d'extracció de característiques, concretament es modifiquen paràmetres corresponents al descriptor objecte d'estudi, tal i com s'ha exposat que es pot fer a l'apartat 3 d'aquest document. En aquest cas, modificant els límits freqüencials del descriptor ens bastarà. Així doncs quedarà loLimit=0Hz (la intenció era canviar-ho a 50Hz però MPEG7 Audio Encoder no ho permet) i hiLimit = 300Hz. El resultat un cop modificat el paràmetre loLimit és el següent:

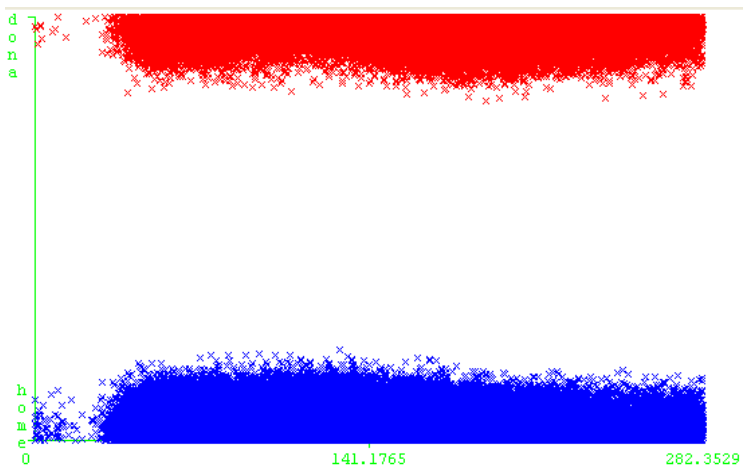


Figura 5.30 Visualització dels resultats de classificació amb el descriptor Audio Fundamental Frequency

Visualment, gairebé no es pot apreciar distinció entre les dues etiquetes. Tot i això es pot veure un lòbul cap a les baixes freqüències en el cas dels homes i un d'altes freqüències en el cas de les dones. Per veure amb més claredat l'aportació d'aquesta característica i cóm es podria tractar-la per obtenir millors resultats es visualitza els resultats corresponents a aquest descriptor per una veu masculina i femenina qualsevol i es representa l'evolució temporal de la freqüència fonamental.

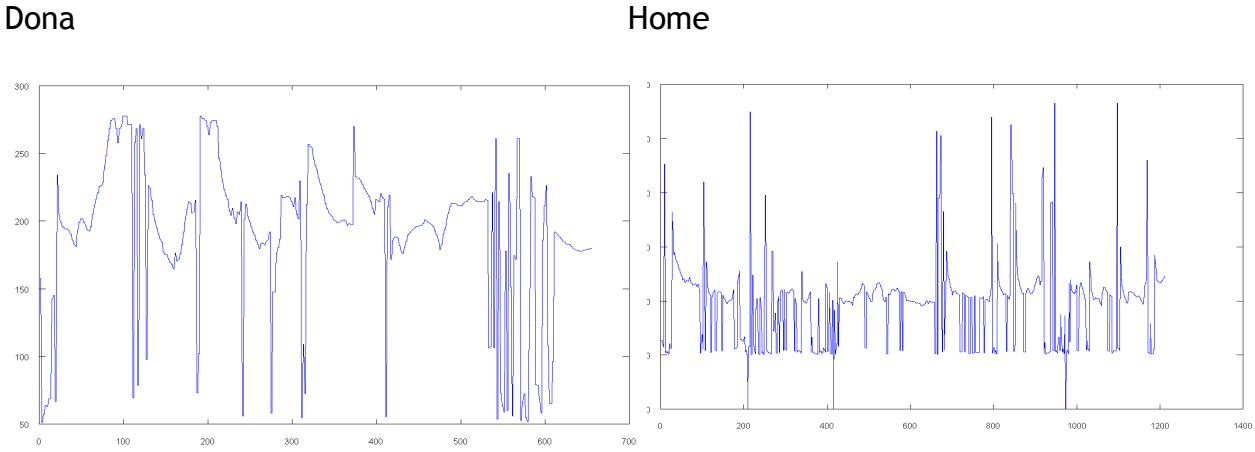


Figura 5.31. Evolució temporal del descriptor Audio Fundamental frequency després de limitar

A la figura anterior es poden veure les dues representacions temporals. A simple vista sembla un resultat erroni, ja que l'evolució de la freqüència fonamental al llarg del temps ha de ser més continua i sense salts tant bruscos. Si ens fixem amb deteniment, però, hi ha un patró de variació que correspon a la freqüència fonamental i que situa la mitjana al voltant dels 200Hz en el cas de la dona i 100 en el cas de l'home.

Un dels possibles motius d'aquest error pot ser la gran quantitat de mostres que es disposen d'un segment determinat ja que es mostreja cada 10ms, és a dir el paràmetre *hopsize*. D'aquesta manera, fonemes amb un alt component freqüencial no permeten obtenir la freqüència fonamental correcte.

Per tant una possible solució serà filtrar pas baix el senyal per tal d'eliminar variacions ràpides en el senyal que no són pròpies de la freqüència fonamental i contribueixen a augmentar l'error. El filtre que s'utilitzarà queda implementat com a un mètode de java, de la manera següent:

```
static public void lowPassIIR(float[] inAndOut, float factor) {
        for (int i = 1; i < inAndOut.length; i++) {

            inAndOut[i] = factor*inAndOut[i] + (1-factor)*inAndOut[i - 1];
        }
    }
```

El resultat de passar les dades per aquest filtre, es mostra a la figura següent, on es representa l'evolució temporal dels dos fragments mostrats anteriorment.

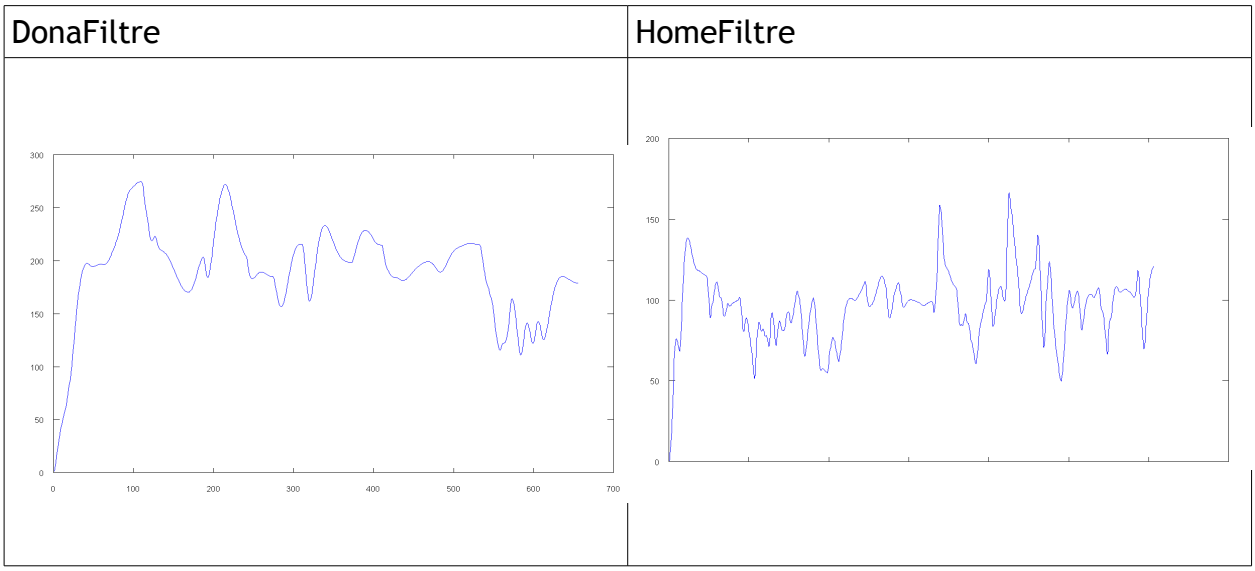


Figura 5.32. Evolució temporal del descriptor Audio Fundamental frequency després de filtrar

Per tal de visualitzar millor el resultat, dibuixem els histogrames corresponents a les dades de sortida del filtre. Com es pot observar a la figura, ara sí que s'observa una clara diferència entre la freqüència fonamental corresponent a una veu masculina amb la d'una veu femenina.

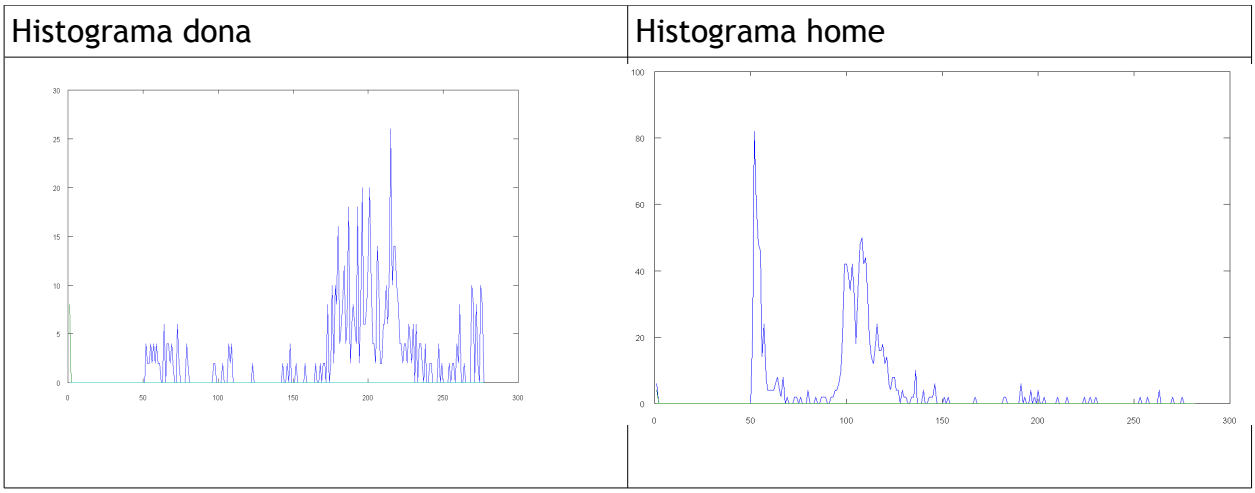


Figura 5.33. Histograma del descriptor Fundamental Frequency

Un cop filtrat el senyal, es procedeix de la mateixa manera que anteriorment quan s'estudiava de manera general la classificació amb totes les etiquetes, i es delma el la freqüència fonamental per un factor de 10. Ara els resultats gràfics de classificació són els que es mostra a la figura, on es veu més clarament una concentració dels valors en un interval de punts per a cada etiqueta.

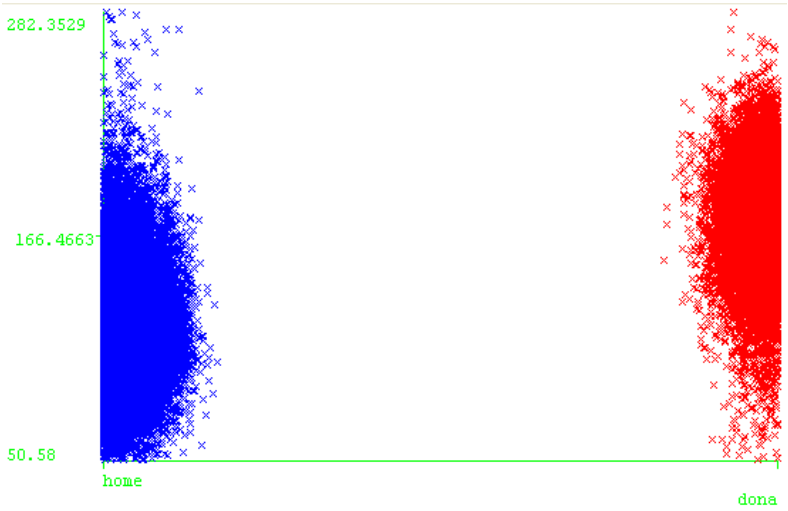


Figura 5.34. Visualització dels resultats de classificació amb el descriptor Audio Fundamental Frequency després de filtrar

Així doncs, ara es el moment de tornar a carregar les noves dades filtrades i delmades al programa Weka i veure els resultats que s'obtenen. Es configura el paràmetre de profunditat amb 1. Pel que fa als resultats del test es pot apreciar un percentatge d'encert del 89%. A la figura següent es pot veure la matriu de confusió i algunes estadístiques. Així mateix, es pot accedir a l'arbre de classificació resultant:

```
Correctly Classified Instances       32186               89.9427 %
Incorrectly Classified Instances      3599               10.0573 %
```

```
=== Confusion Matrix ===

     a     b   <-- classified as
 27571   655 |     a = home
  2944  4615 |     b = dona
```

Figura 5.35. Valors obtinguts amb el arbre classificador

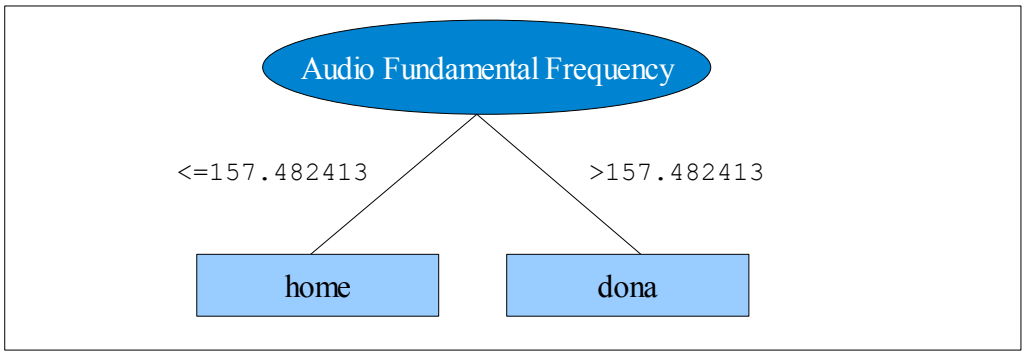


Figura 5.36. Arbre de classificació per al descriptor Audio Fundamental Frequency

Així doncs el resultat serà un únic llindar el qual quedarà marcat als 157,48 Hz. La implementació d'aquest arbre es pot fer mitjançant sentències condicionals. D'aquesta manera, la implementació serà la següent:

```
if(AFFvalor<=/*157.482413*/145.72){ Etiqueta="home";}
else { Etiqueta="dona";}
```

Tot i el bon funcionament d'aquest descriptor es continuaran estudiant els altres descriptors a fi de millorar al màxim la classificació.

ESTUDI DEL DESCRIPTOR "AUDIO POWER"

En aquest punt s'estudiarà la potència de l'àudio. A priori sembla que aquest paràmetre no hauria de ser molt significatiu ja que en realitat només expressa la variació de potència en funció del temps i aquesta descripció podria ser utilitzada per tal de diferenciar altres etiquetes les quals no són motiu d'estudi, però que podrien ser veu cridada, veu parlada, etc.

Igualment que amb el descriptor anterior, s'introdueixen les dades a Weka sense cap tipus de pre-processat.

```
    a     b   <-- classified as
45884     0 |     a = home
12134     0 |     b = dona
```

Figura 5.37: Matriu de confusió pel descriptor Audio Power

Els resultats són molt dolents, no es troba un llindar que diferenciï les dues etiquetes. De manera anàloga a com s'ha procedit anteriorment, s'estudia temporalment un fragment corresponent a un home i a una dona. L'evolució temporal es mostra a la següent figura, combinat amb un filtre igual a l'anterior.

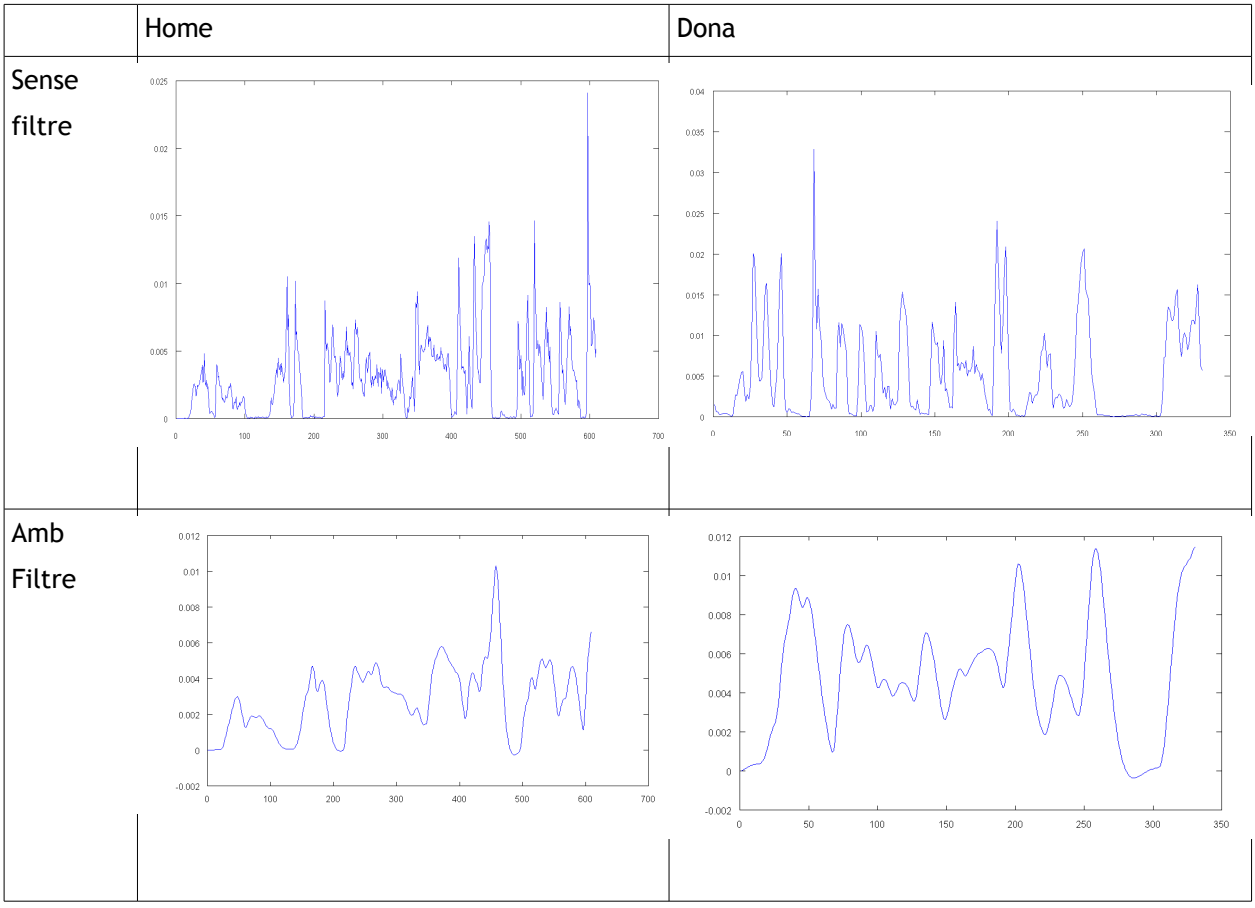


Figura 5.38. Evolució temporal de les dades corresponent a Audio Power sense filtrar i filtrades

Tot i filtrar les dades, visualment no s'aprecia una diferència considerable entre les dues etiquetes pel que fa als valors del descriptor Audio Power. Així doncs, es considera que no es rellevant la informació que aporta aquest descriptor.

ESTUDI DE LA RESTA DE DESCRIPTORS

S'estudien tots els descriptors restants de la mateixa manera que s'ha procedit anteriorment. El resultat d'aquest estudi, no mostra cap altre classificador que presenti una rellevància tant elevada com Audio Fundamental Frequency. Per tant, es decideix elaborar el classificador home-dona només amb un descriptor: Audio Fundamental Frequency.

5.3 Implementació del software

Als apartats anteriors, s'ha estudiat el problema per tal d'aconseguir estudiar les característiques d'una base de dades i establir uns criteris o models de classificació. Aquesta primera fase és la fase d'entrenament. A partir d'aquí queden una segona i una tercera fase per realitzar. La segona fase és la fase en què s'implement el software de classificació automàtic. I la tercera fase, es la fase de test, que es veurà a l'apartat 6 d'aquest document.

Així dons, un cop queda definida la classificació de manera individual, s'està en disposició de totes les eines necessàries per elaborar un sistema de classificació genèric i automàtic.

Amb el treball fet anteriorment, el sistema és capaç de classificar un segment de donat amb una sola etiqueta. L'objectiu per contra, no només és assignar un valor a un segment, sinó que és trobar dins d'un arxiu els diferents canvis d'esdeveniments sonors i classificar-los de manera correcta.

A partir dels algorismes que es presenten als punts següents, es pretén assolir aquest objectiu, on es tractaran els classificadors específics per a cada problema de manera general i s'automatitzarà tot el procés d'extracció de característiques i classificació segons la següent figura.

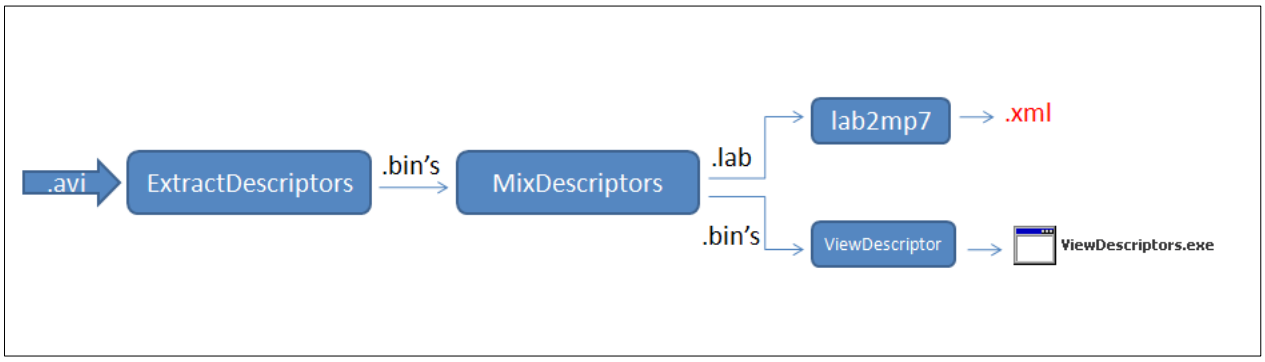


Figura 5.39. Procés de classificació automàtica

Així doncs, aquest procés consta de tres passos , per una banda ha de permetre extreu-re els descriptors de l'àudio extret del vídeo d'entrada d'una manera específica, en segon lloc, tractar els descriptors i classificar els segments, i finalment, visualitzar els resultats de diferents maneres.

5.3.1. Extracció de les característiques i classificació: Extract Descriptors i MixDescriptors

En aquest punt, es parlarà del procés de classificació automàtica pròpiament dit, és a dir, de tots els passos que s'han de seguir per tal d'obtenir una sortida segmentada i classificada. Cal dir abans d'endinsar-se massa en l'explicació de l'algorisme, que es troba format per diferents funcionalitats implementades anteriorment que han servit també per la fase de training. Així doncs, pel tal de facilitar l'enteniment, la figura 5.36 mostra els processos i les transformacions que sofreixen les dades durant tot el procés.

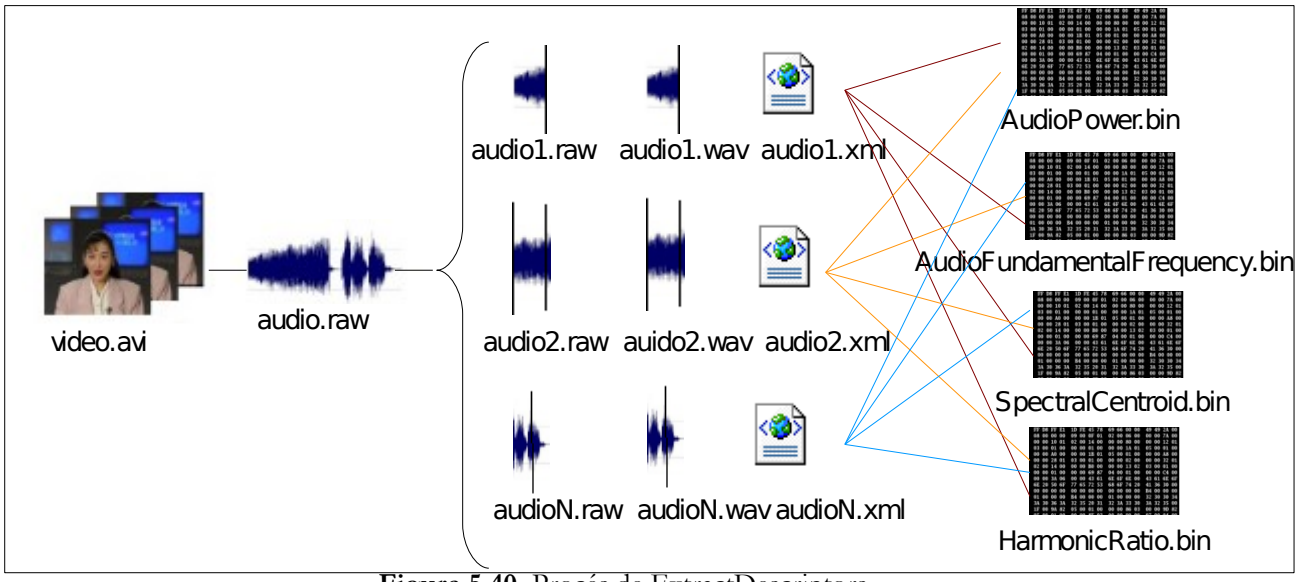


Figura 5.40. Procés de ExtractDescriptors

L'única entrada al programa ExtractDescriptors és el vídeo d'entrada, a part, han d'estar indicats les rutes d'accés al vídeo, al programa ffmpeg i al Mpeg-7 Audio Encoder. El primer pas és extreure l'àudio del vídeo d'entrada de la mateixa manera que s'ha realitzat en apartats anterior, a partir del programa ffmpeg. En aquest cas l'àudio s'extreu en format Raw, és a dir, la informació en brut. A partir del mètode splitRaw, es trosseja l'àudio en segments de duració dos segons. Per tant ara es tenen tants segments en format Raw com la duració total de l'àudio entre dos.

Quan aquest procés ha finalitzat, es crea segment per segment la seva còpia en format wav per tal de poder passar aquest arxiu com a entrada al Mpeg-7 Audio Encoder, i així crear la descripció segons l'arxiu de configuració on s'especifiquen quins descriptors s'utilitzen. Per aquest procés l'arxiu de configuració pot estar compost només pels descriptors significatius escollits a la fase d'entrenament, ja que són els que realment s'utilitzaran per la classificació. Tot i això, s'escollirà una configuració amb tots els descriptors seleccionats i amb el retall de *hiLimit* i *loLimit* al descriptor AudioFundamentalFrequency tal i com s'ha explicat a l'apartat 5.2.2.1.

D'aquesta manera, quan es crea la descripció d'un segment, s'insereix a un arxiu binari per a cada descriptor que hi ha, és a dir, cada segment tindrà la seva aportació a AudioPower.bin, AudioFundamentalFrequency.bin, etc. En conseqüència l'accés al final serà més senzill i l'espai que s'ocupa en disc també disminueix.

A partir de les arguments *create, delete,* es permeten seleccionar diferents opcions en cas de que l'àudio audio.raw ja existeixi, eliminar-lo i crear un de nou, o no eliminar-lo i no crear un de nou. L'argument *deleteIntermediate* per la seva banda, permet eliminar tots els arxius creats que només son intermediaris, com ara tots els arxius raw segmentats, els seus arxius en format wav corresponents, i la descripció en xml, ja que tota aquesta informació no serà necessària un cop creats els arxius binaris.

Un dels motius d'aquest procediment resideix en el cost temporal que presenta els programes ffmpeg i Mpeg-7 Audio Encoder, reduint-se si es segmenta manualment l'àudio extret en format raw en comptes de treballar com en algorismes explicats per realitzar apartats anteriors, on la instrucció que cridava ffmpeg ja demanava al programa la segmentació segons els intervals inicial i final de duració. A més, pel que fa als descriptors del ScalarType, retornen un sol valor, això vol dir que per tot l'arxiu d'àudio només es tindria un valor representatiu per aquests descriptors. Clarament, això no valdria per a classificar un arxiu on apareixen diferents esdeveniments sonors, i la

manera que es té per aproximar és segmentar l'àudio aleatòriament, en aquest cas, cada dos segons. Així doncs, els valors corresponents als descriptors ScalarType s'obtenen cada 2segons. Ja que volem arxius binaris de la mateixa longitud, la tècnica utilitzada per escriure aquests binaris serà repetir el mateix valor cada 10ms durant 2segons. Així doncs, en finalitzar l'execució d'aquest programa, es disposarà de tants arxius binaris com descriptors s'hagin seleccionat a l'arxiu de configuració del Mpeg-7 Audio Encoder, els quals descriuen el contingut de l'àudio extret del vídeo d'entrada.

Pel que fa al programa MixDescriptors, es llegeixen els binaris generats anteriorment, havent especificat la ubicació d'aquests, i s'emmagatzema la informació de cadascú en un vector corresponent.

Així doncs, disposem de vectors amb una mostra casa 10ms, tal i com s'especifica a la configuració del Mpeg-7 Audio Encoder. Pel que fa a la classificació es defineix un arbre de la mateixa manera que a s'ha definit a l'apartat anterior. Primerament es discrimina entre silenci o no-silenci, en cas de no-silenci, es discrimina entre música o veu, i finalment, en cas de veu es discrimina entre home i dona. En aquest cas, tal i com mostra la figura 5.37, a aquestes etiquetes se li assignen uns valors, numèrics, essent 0 silenci, 1 música, 2 home i 3 dona. Aquesta classificació es genera cada 10ms i s'emmagatzema a un arxiu binari, de la mateixa manera que també s'emmagatzemen les classificacions parcials.

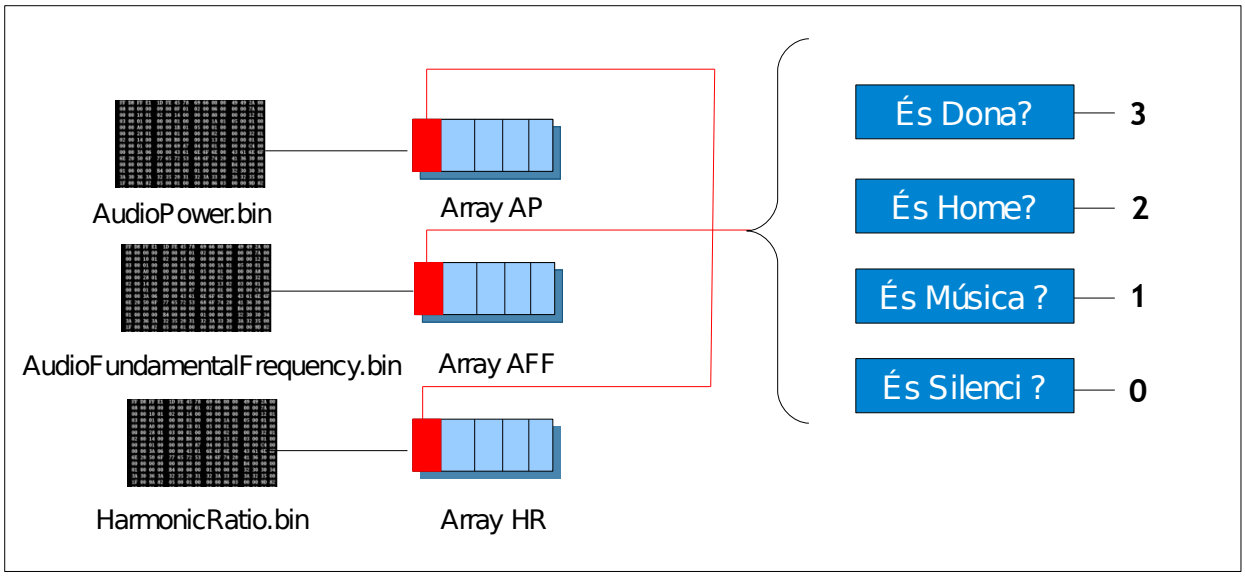


Figura 5.41. Procediment que es segueix a MixDescriptors

La sortida del classificador general però, no es realitza cada 10ms. Definim per tant, un valor de limitació de la sortida de les dades del classificació. Aquest valor és *nMitja,* i és configurable dins del codi font corresponent a MixDescriptors. Així doncs, la sortida serà la mitjana aritmètica entre

els valor a determinar per *nMitja*. Per defecte aquest valor queda configurat a 50, de manera que la limitació serà de 50 mostres · 10 ms / mostra = 500 ms = 0.5 segons. El altres paraules, el classificador només serà capaç de detectar canvis cada mig segon.

En acabar l'execució de MixDescriptors, disposarem d'un conjunt d'arxius binaris corresponents a tots els descriptors, les classificacions parcials, i la classificació final.

5.3.2. Representació de la classificació i segmentació: ViewDescriptors i MixDescriptors

Ara ja es disposa de la classificació pròpiament dita per a un vídeo determinat. El que queda per fer es mostrar les dades d'alguna manera. S'ha decidit mostrar les dades de dues maneres, textualment i gràficament. Ja que a més a més, es necessita algun arxiu amb els resultats per tal de realitzar les proves que s'exposaran a l'apartat 6, la manera de fer-ho serà idèntica a la que s'ha utilitzat a l'etiquetat manual, format lab.

Així doncs, s'incorpora al programa MixDescriptors la funció seleccionable o no de crear l'arxiu lab. Aquest es crea trobant les transicions i calculant el temps inicial i final de cada etiqueta. A més també es disposa de l'opció de crear l'arxiu de descripció mpeg-7 (.xml) corresponent.

Pel altra banda, i per requeriments de l'empresa, es volen visualitzar les dades de manera visual i simultània al vídeo d'entrada. Així doncs, un cop executats els programes ExtractDescriptors i MixDescriptors, es pot executar ViewDescriptors i així iniciar la visualització. Aquesta interfície té la següent estructura:

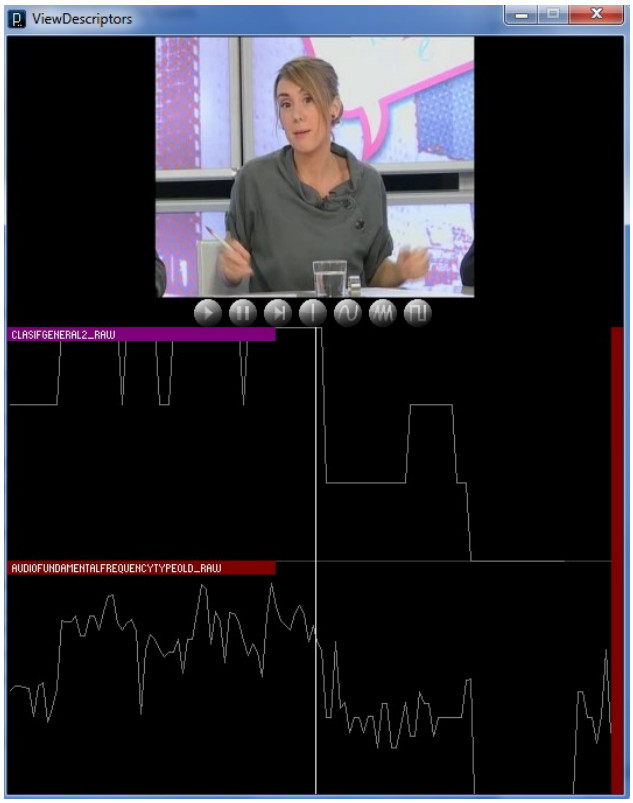


Figura 5.42 Captura de pantalla de l'aplicació ViewDescriptors

L'aplicació ha estat desenvolupada sota la plantilla que l'empresa VSN ens va facilitar, i per defecte presenta l'estructura de la figura anterior. Per tal de seleccionar el vídeo d'entrada cal modificar un arxiu de configuració que ha de residir a C:/ amb el nom *args.txt*. La informació que ha d'haver-hi a aquest arxiu és la següent:

- Ubicació dels arxius binaris creats per ExtractDescriptors i ViewDescriptors

- Ubicació del vídeo

- Número de gràfiques que volem per pantalla

Exemple:

```
c:/projecte/AVI/
c:/projecte/AVI/banda1.avi
4
```

Per defecte, el nombre de representacions que apareixen per pantalla són dos. Les funcionalitats que permet el programa són les que es mostren a la figura 5.39:

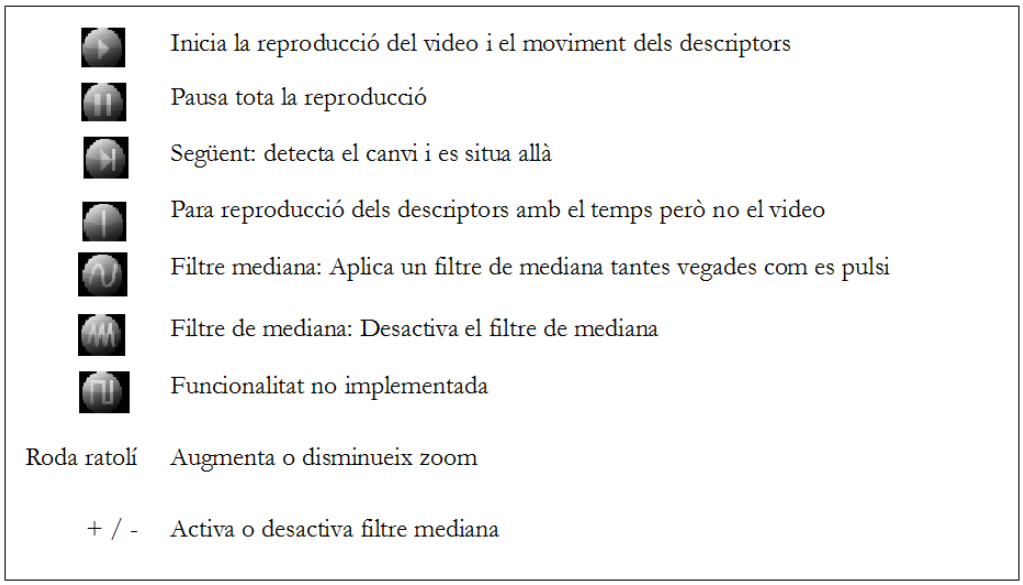


Figura 5.43. Taula amb les funcions que permet el programa ViewDescriptors

La informació es mostra a partir de llegir els arxius binaris i representar els seus valors com una gràfica. Aquests arxius binaris que es poden llegir seran tots els que s'han creat anteriorment amb els programes ExtractDescriptors i ViewDescriptors. Així doncs, es podran visualitzar els descriptors extrets amb la seva evolució temporal, els classificadors parcials, i el classificador final, el qual te els següents nivells.

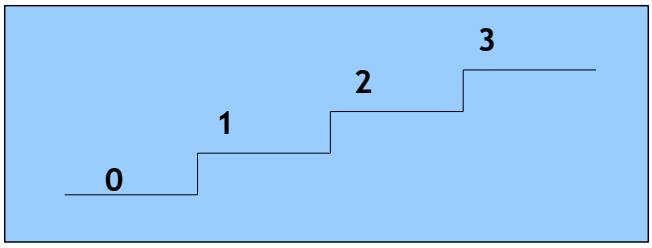


Figura 5.44. Estructura de la representació de les dades del classificador general.

5.4 Esquema general del sistema

Així doncs, el procés de classificació queda finalitzat. A la figura 5.45 es mostren tots els processos explicats anteriorment i les seves relacions.

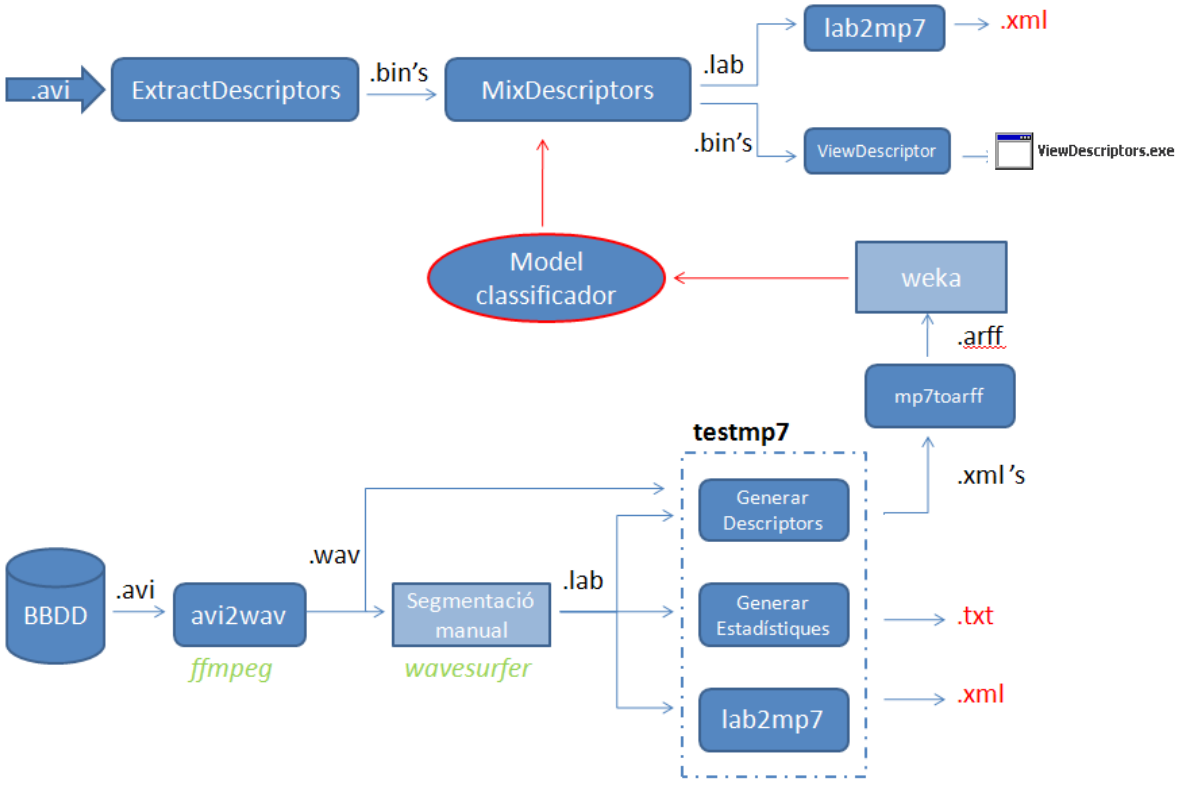


Figura 5.45. Esquema de visió general de tot el procés d'entrenament i classificació

6. Experiments i resultats

Aquesta part del projecte contempla el testeig del sistema classificador dissenyat en l'apartat anterior. Els resultats d'aquest procés, representat en la figura 6.1, ens donaran una idea de quant be fa la classificació el nostre sistema, dels punts més febles que té i ens podrà guiar en cas de resultats negatius cap a on hem d'encarar les millores del sistema.

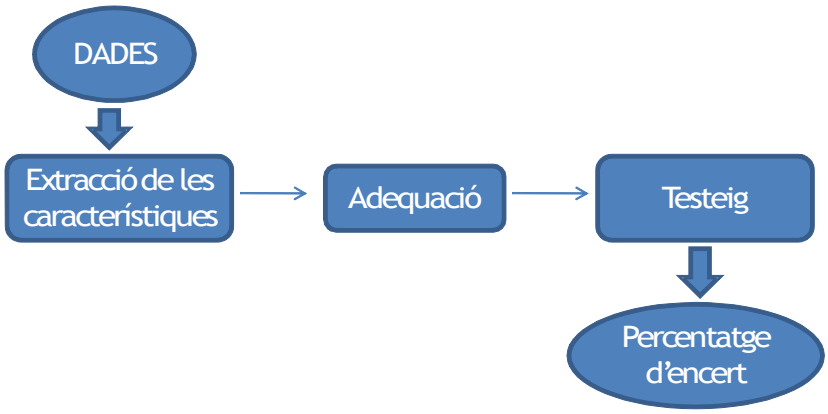


Figura 6.1. Esquema general del procés d'avaluació.

Per tal de fer el testeig el més acurat possible, ho hem dividit en dos parts separades. Un primer testeig avalua cada model classificador per separat, de forma que ens deixa veure l'eficàcia de la classificació. En un altre procés avaluem l'eficàcia del sistema a l'hora de determinar els punts de canvi, les separacions entre segments. És per tant una avaluació de l'eficàcia de la segmentació.

Per a portar a terme aquests experiments, s'ha utilitzat la part de la base de dades destinada a test: l'última porció de set dies i el programa naturalment10, a més d'incloure un nou programa de banda i banda, el banda3.

6.1 Mètode d'avaluació de la classificació

Per tal d'avaluar el comportament dels nostres classificadors, hem dissenyat un algoritme en llenguatge JAVA. L'introduïm un seguit de segments, que prèviament hem creat i etiquetat manualment, i fem que el classificador els etiqueti. Finalment, comparem l'etiqueta de sortida amb la d'entrada. Hi ha hagut un petit problema, però. Degut a que cada classificador utilitza uns descriptors i un sistema de classificació diferents, l'algoritme també l'hem hagut d'adequar a cada classificador.

En el cas dels discriminadors de gènere i de silenci/no silenci, que utilitzen descriptors que retornen un valor cada 10 ms, l'algoritme treu en resultat cada 10 ms. Després es fa un percentatge sobre el total de mostres de cada segment. L'etiqueta que té major percentatge (superior al 50%) és la que es considera bona. En el cas del discriminador veu/música, obtenim un resultat directament per a cada segment, ja que utilitzem descriptors que retornen un escalar, i per tant es directament aquesta etiqueta la que es compara amb l'etiqueta d'entrada.
Finalment, es fa un percentatge final d'encerts.

6.2 Mètode d'avaluació de la segmentació

També s'ha dissenyat l'algoritme que avalua la segmentació, en aquest cas basant-se en el mètode utilitzat a[13] . Podem trobar-nos amb dos tipus d'error, quan s'avalua la segmentació. Un primer error seria que el sistema detecti un canvi que en realitat no ha ocorregut i s'anomena recall (R). L'altre, succeeix quan un veritable canvi d'esdeveniment sonor no es detectat per el sistema, i s'anomena precission (P).

Per tant, es podria considerar el precission com la proporció de transicions detectades pel sistema automàtic que concorden amb transicions marcades manualment. Així doncs, el recall seria la proporció de transicions manuals que concorden amb una detectada automàticament. Si considerem "A" com les transicions detectades pel sistema automàtic, "M" les detectades manualment i "C" com les que concorden d'ambdues mesures, llavors els valors de precission i recall es formulen com:

$$P = C/A$$

$$R = C/B$$

També és necessari determinar un altre paràmetre, que anomenat "W". Aquest valor serveix per a l'hora de determinar P i/o R, fixar una franja dins la qual es poden prendre dues transicions (una d'automàtica i una altra manual) com coincidents. Els valors típicament utilitzats en mesures d'aquest tipus oscil·len entre 0,5 i 3 segons. Un exemple el veiem a la figura 6.2. La franja groga representa el valor de "W". Les transicions, representades per línies, que coincideixen són les pintades de negre mentre que les que no coincideixen estan en vermell. D'aquesta forma, el valor de precission seria de 3/6 i el de recall de ¾. La millor valoració es el 1 metre que la pitjor en un zero.

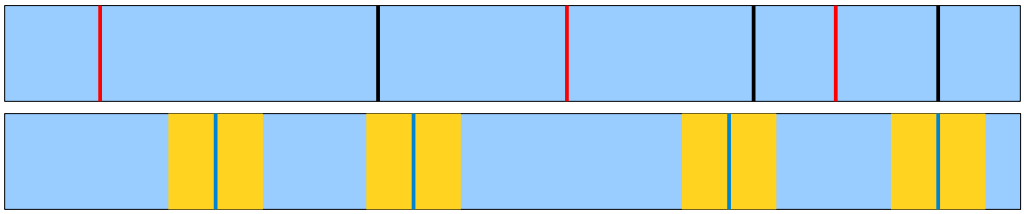


Figura 6.2 Exemple d'avaluació de segmentació. La línia de dalt són les transicions automàtiques i la de sota les manuals

Dins del sistema exposat en aquest projecte, en el cas d'avaluar la segmentació es necessita un arxiu complet, amb les seves corresponents transcripcions, la manual i la generada automàticament a la sortida del programa MixDescriptors.

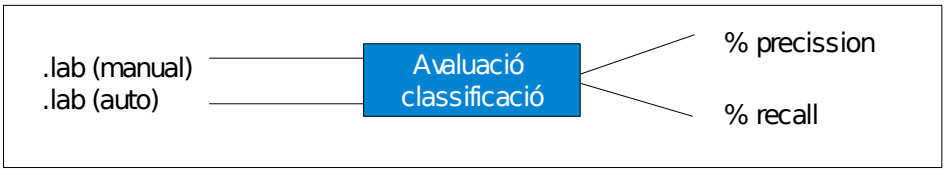


Figura 6.3. Esquema de l'avaluació de la segmentació

Abans, però cal adequar els arxius de transcripció manual per tal de realitzar correctament l'avaluació. Els arxius obtinguts en fer la segmentació manual estaven composats per unes etiquetes que finalment no han estat les assignades automàticament. És a dir, a la segmentació manual es tenia en compte alguns matisos que no s'han tingut en compte en la classificació automàtica, com ara.

- Canvis de locutor

- Canvis de peces musicals

- Veus que no son pures (Ex. HomeSoroll, donaMusica, etc.)

Per tal de compensar aquestes mancances als arxius de transcripció generats manualment, es

realitzen algunes transformacions per tal d'eliminar els matisos explicats anteriorment. Així doncs, ara es disposa d'uns arxius de transcripció (.lab) generats de manera automàtica i uns altres generats de manera manual però adequats a la situació que es presenta.

6.3 Avaluació dels classificadors

6.3.1 Classificador silenci-no silenci

Hi ha hagut un problema a l'hora d'avaluar aquest classificador i és l'ínfim nombre de segments de silenci que apareixen als arxius de la base de dades utilitzats en el procés de test. Per tal de poder fer l'avaluació, s'ha decidit generar manualment segments de silenci mitjançant el programa d'edició de so Adobe Audition. En total s'han generat cinquanta segments .wav mono a 44100 Hz i 16 bits de silenci de diferents durades, afegint a cadascun un soroll que podia ser marro, rosa o blanc, a diferents intensitats entre 2 (el mínim que permet el generador) i 3. El fet d'escollir aquests valors es que s'ha comprovat que per sobre de 3 el to ja era prou alt i els valors del AudioPower del segment generat es trobaven sobre el marge del que el nostre sistema considera silenci.

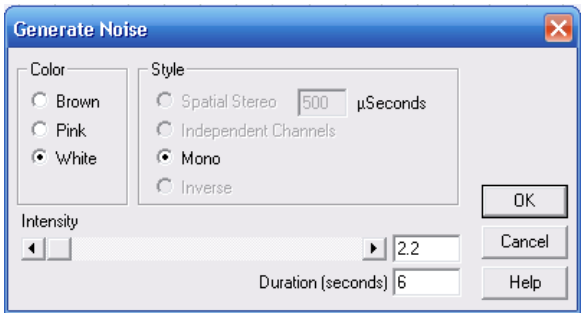


Figura 6.4. Captura de pantalla del menú de generació de soroll d'Audition.

Les descripcions d'aquests segments han sigut introduïdes al sistema avaluador conjuntament amb 50 mostres d'àudio de la base de dades agafades a l'atzar. Els resultats es mostren a continuació.

TOTAL		
Silenci	No silenci	
48	2	Silenci-no
0	50	No silenci

Figura 6.5. Matriu de confusió corresponent a les proves del classificador silenci-no silenci.

6.3.2 Classificador veu-musica

A continuació es mostraran les proves de classificació de veu i música. Es disposa d'una base de dades destinada a proves, la qual s'ha dividit per realitzar les diferents proves. Els resultats de realitzar les proves de discriminació de veu i música a la base de dades a la qual només hi figuren arxius corresponents a veu i música, són els representats a la figura 6.4.

Música	Veu	
30	18	Música
1	28	Veu

Música	Veu	
0	1	Música
5	82	Veu

Música	Veu	
0	0	Música
0	44	Veu

Total		
Música	Veu	
30	19	Música
6	154	Veu

Figura 6.6. Matrius de confusió corresponents a les proves per a discriminar veus i música.

Com es pot observar, el percentatge d'encert es prou elevat, obtenint un global al voltant del 88%. No obstant, s'aprecia masses problemes a l'hora de classificar les músiques. Degut a la voluntat de perfeccionar aquests resultats, s'han fet una serie d'experiments.

EXPERIMENT 1A

Per tal de millorar la classificació de la música, primer s'ha avaluat si utilitzant només el descriptor que millors resultats ha donat a l'entrenament, el HSD, que aconsegueix un percentatge del 70% en músiques, és copsen millors resultats. El model classificador s'ha extret de weka utilitzant com a únic paràmetre d'entrada HSD.

TOTAL		
Música	Veu	
18	30	Música
0	160	Veu

Figura 6.7. Matrius de confusió corresponents a l'experiment 1A

Els resultats no son satisfactoris. El percentatge baixa al 85 %, i tot i que sembla classificar millor les veus, es perd qualitat a l'hora de classificar la veu, objectiu que es volia assolir.

EXPERIMENT 1B

Ja que l'experiment anterior ha fracassat, s'avalua el classificador agafant els dos millors de l'entrenament segons el model mostrar a la figura 6.8, amb els resultats exposats a la figura 6.9:

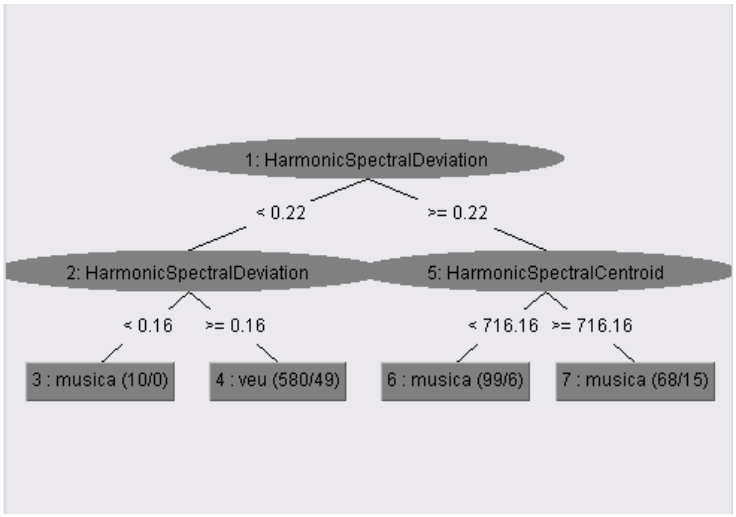


Figura 6.8 Model de classificació veu-música per a l'experiment 1B

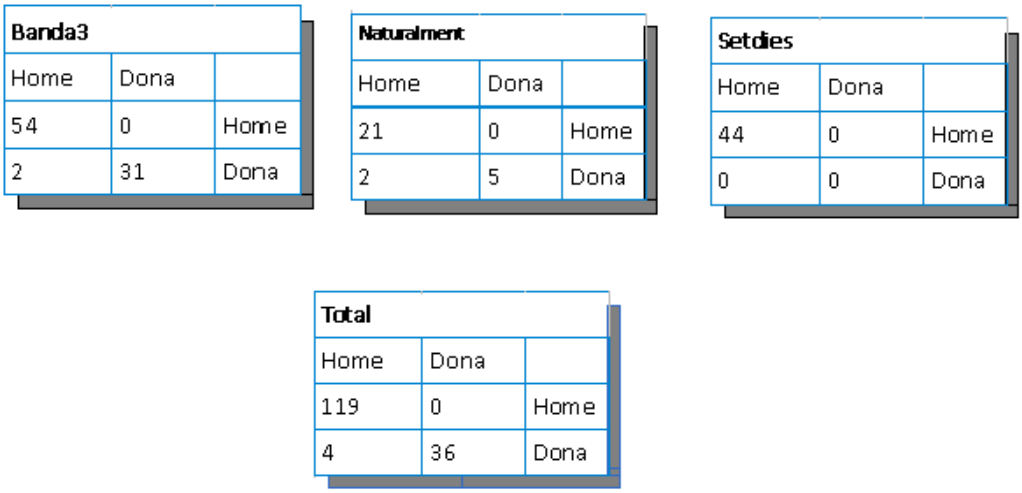
TOTAL		
Música	Veu	
20	28	Música
73	87	Veu

Figura 6.9. Matrius de confusió corresponents a l'experiment 1B

El resultat es encara pitjor, com era previsible, amb un encert general del 54%. No sembla que es pugui trobar millora amb aquests descriptors.

6.3.3 Classificador home- dona

Els resultats de realitzar les proves de discriminació de home i dona a la base de dades destinada a la qual només hi figuren arxius de veu corresponents a homes i dones, són els representats a les següents matrius de confusió.



Banda3

Home	Dona	
54	0	Home
2	31	Dona

Naturalment

Home	Dona	
21	0	Home
2	5	Dona

Setdies

Home	Dona	
44	0	Home
0	0	Dona

Total

Home	Dona	
119	0	Home
4	36	Dona

Figura 6.10 Matrius de confusió corresponents a les proves per a discriminar homes i dones

Les proves realitzades per a aquest classificador donen un valor d'encert general del 97%, per tant, són uns resultats molt bons. Tot i això, es decideix provar altres arbres classificadors on és vegin implicats més descriptors de baix nivell. Els resultats però, no mostren una millora considerable. Així doncs, es decideix no modificar aquest classificador i utilitzar la configuració adoptada primerament.

6.4 Avaluació de la segmentació general

A continuació es realitzaran les proves pels diferents arxius que es disposen a la base de dades: banda3, setdies3, naturalment8. Les proves realitzades per a cada programa són per a una sensibilitat d'un segon, de dos segons, i finalment de tres segons. Els resultats obtinguts per "Banda3", "Setdies3" i "Naturalment10" es presenten a les figures 6.5, 6.6 i 6.7 respectivament.

Banda3	**Precission (%)**	**Recall (%)**
Sensibilitat 1s	14	61
Sensibilitat 2s	21	91
Sensibilitat 3s	21	94

Figura 6.11. Resultats per l'avaluació de l'arxiu "banda3"

Setdies3	**Precission (%)**	**Recall (%)**
Sensibilitat 1s	2,1	50
Sensibilitat 2s	2,1	50
Sensibilitat 3s	2,1	50

Figura 6.12. Resultats per l'avaluació de l'arxiu "Setdies3"

Naturalment10	**Precission (%)**	**Recall (%)**
Sensibilitat 1s	15	45
Sensibilitat 2s	23	67
Sensibilitat 3s	27	78

Figura 6.13. Resultats per l'avaluació de l'arxiu "Naturalment10"

Pel que fa als resultats de l'avaluació de la segmentació, s'observen diferents punts d'error. En primer lloc, es veuen uns tants per cent del paràmetre "precission" molt baixos en tots els casos, però especialment al programa "Setdies3". Aquest problema és degut a que el sistema classificador presenta molts canvis d'esdeveniment on veritablement no hi ha, i com es mostra a la figura 6.14, on la part superior correspon al segment original i la part de sota correspon a la sortida del classificador.

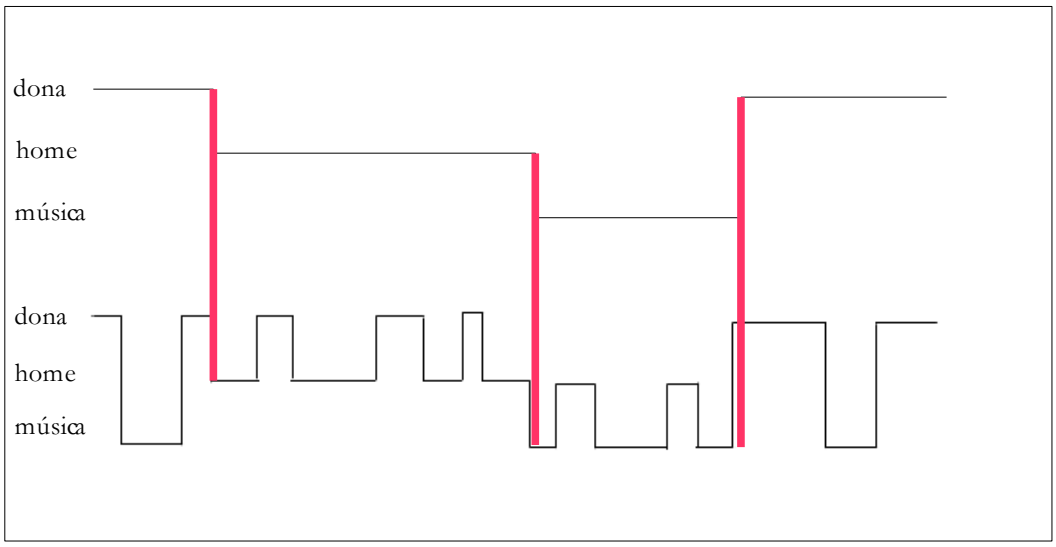


Figura 6.14. Exemple avaluació paràmetre "precission". Part superior original part inferior sortida classificador

El principal problema i el que més distorsiona el valor de "precission" és degut als resultats dolents que presenta el classificador veu-música, tot i que també el classificador home-dona presenta més errors a l'hora d'avaluar un arxiu sencer que no pas a l'avaluació realitzada anteriorment.

En el cas de tenir errors entre classificació d'home i dona, aquest error es podria disminuir aplicant el filtre de mediana implementat a l'aplicació ViewDescriptors ja que degut a les característiques del sistema, els pics corresponents a l'error són de mig segon. Així doncs, aplicant aquest filtre s'eliminen els pics i es perd resolució temporal degut a les característiques del filtre: no linealitat i no casualitat.

En el cas de veu i música, les característiques del classificador fan que els errors mínims tinguin una duració de dos segons. Per tant aquests errors no es poden reduir a partir del filtre de mediana.

7. Conclusions

Un cop finalitzat tot el procés del projecte, és moment de la interpretació dels resultats obtinguts. Tal i com s'ha pogut apreciar als apartats 5 i 6 d'aquest document, s'ha dissenyat un sistema classificador automàtic que diferència entre diferents classes bàsiques i s'han obtingut els resultats corresponents a diferents proves que s'han efectuat. Així doncs, s'ha assolit aquest objectiu general. Pel que fa a l'eficiència del classificador, s'esperaven millors resultats ja que les referències principals a l'hora d'elaborar el projecte presentaven uns tants per cent d'eficiència lleugerament superiors als presentats en aquest projecte.

En referència a l'eficiència del classificador pròpiament dit, els resultats obtinguts a l'apartat 6.3 són força bons en general. Tal i com s'ha explicat anteriorment, aquestes proves s'han realitzat a arxius ja segmentats manualment i la prova realitzada era classificar individualment els arxius. A l'apartat 6.4 en canvi, s'avalua l'eficiència general del sistema, és a dir, l'arxiu d'entrada és un vídeo gran on apareixen diferents esdeveniments sonors i s'ha de segmentar i classificar de manera simultània. És en aquest cas quan l'error augmenta, detectant molts canvis quan no existeixen realment i els canvis detectats de tant en tant els classifica de manera errònia degut a que la finestra de dos segons que s'utilitza pot abastar més d'un esdeveniment sonor.

Amb una visió més de cara a l'empresa aquests resultats no són del tot dolents, ja que el principal error és la detecció de massa canvis quan no existeixen realment (precission), però en cas de detectar un canvi correcte (recall), aquest té un tan per cent d'encert bastant més elevat. Aquest era doncs, el principal requeriment: detectar canvis.

Pel que fa als objectius no assolits, es troben la discriminació de diferents locutors dins del mateix genere i la discriminació de diferents peces musicals. Aquests objectius no han estat assolits degut principalment a falta de temps per un projecte tan ambiciós.

Així docs, amb una mica més de perspectiva, es planteja el dubte de si realment els descriptors de baix nivell Mpeg-7 utilitzats exclusivament presenten un bon funcionament i a més, si l'esquema adoptat va ser el correcte, és a dir, realitzar una segmentació i classificació simultània.

7.1 Futures línies de treball

Les principals línies de treball que es proposen són esmenar els problemes mencionats anteriorment i tractar d'assolir els objectius no assolits en aquest projecte. Es proposen els següents treballs:

- Prova de realitzar un arbre classificador amb altres descriptors
- Altre plantejament general del projecte amb una segmentació inicial utilitzant la tècnica BIC Segmentation
- Prova de diferents algorismes de classificació com HMM, NNR...

Un cop assolits aquests objectius de millora del classificador actual, pot ser moment de implementar noves funcionalitats com ara:

- Discriminació de diferents peces musicals i diferents locutors
- Identificació de locutors.

7.2 .Opinió personal

En general aquest projecte ens ha comportat una càrrega de treball molt gran. Tot i que els resultats obtinguts no han estat els que ens esperàvem, hem adquirit experiència en desenvolupar un projecte gairebé des de cero a l'entorn de treball que és una empresa. Valorem positivament l'experiència i en general estem satisfets del resultat obtingut.

8. Referències

[1]- http://www.oreillynet.com/onjava/blog/2006/05/explicit_and_implicit_metadata_1.html

[2]- http://www.interacciones.com.ar/metadatos-y-metacrap/

[3]- http://es.wikipedia.org/wiki/Metadato

[4]- Hyoung-Gook Kim, Nicolas Moreau, Thomas Sikora, "MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval" Wiley, October 2005.

[5]- José M. Martínez MPEG-7 Overview (version 10) ISO/IEC JTC1/SC29/WG11N6828 Palma de Mallorca, October 2004

[6]- Català Onaindia, David, "Reconeixement facial amb tècniques mixtes 3D-2D", Projecte Final de Carrera UPC, octubre 2006

[7]- J. Garcia Arnal Barbedo, AES Member, and A.Lopes, "A Robust and Computationally Efficient Speech/Music Discriminator", November 2005.

[8]- Mingsain R. Bai, AES Member, and Meng-chun chen - "Intelligent Preprocessing and Classification of Audio Signals", 2006 July.

[9]- E. Scheirer and M. Slaney, "Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator," in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (Munich, Germany, 1997 Apr.), pp. 1331–1334.

[10]-http://www.tom.comm.waseda.ac.jp/map7/table.html

[11]- Piotr Szczuko, Piotr Dalka, Marcin Dabrowski and Bozena Koster, Audio Engineering Society Convention Paper 6105, " MPEG-7 -based low-level descriptor effectiveness in the automatic musical sound classification", May 2004.

[12]- Mingsain R. Bai, AES Member, and Meng-chun chen - "Intelligent Preprocessing and Classification of Audio Signals", 2006 July.

[13]- Vincenzo Dimattia, UPC: "PFC: An Automatic Audio Segmentation System for Radio Newscast", March 2008.

[14]- Giuseppe Dimattia, UPC: "PFC: An Automatic Audio Classification System for Radio Newscast", March 2008.

[15]- Hyoung-Gook Kim, Edgar Berdahl, Thomas Sikora, "Study of MPEG-7 Sound Classification and Retrieval", October 2005

[16]- MUVIS http://muvis.cs.tut.fi/contact.html

[17]- Paper Muvis

[18]- http://clam-project.org/index.html

[19]-MPEG-7 Audio Encoder - http://MPEG-7audioenc.sourceforge.net/

[20]- http://iiss039.joanneum.at/cms/index.php?id=84

[21]- http://www.sail-technology.com/

[22]- El estàndard... juan jose El estándar MPEG-7 por Pedro José Vivancos Vicente

[23]- 55th Mpeg Meeting- MPEG Requirements Group, "55th MPEG Meeting," MPEG-7 requirementsDocument V.13, Pisa, Italy, Doc. ISO/MPEG N3933, Jan. 2001.

[24]- http://www.xtec.es/~jvivanco/odd/xml_intro.pdf

[25]- Jane Hunter, "An Overview of the MPEG-7 Description Definition Language (DDL)", June 2001.

[26]- http://java.sun.com/javase/downloads/index.jsp

[27]- http://netbeans.org/

[28]- http://www.ffmpeg.org/

[29]- http://www.speech.kth.se/wavesurfer/

[30]- http://www.ient.rwth-aachen.de/cms/

[31]- http://www.rwth-aachen.de/go/id/bdz/

[32]- XMLBeans- http://xmlbeans.apache.org/]

[33]- http://www.cs.waikato.ac.nz/~ml/index.html

[34]- Diego García Morate- "Manual de Weka"

[35]- Bernhard Pfahringer, University of Waikato -"Weka, a tool for exploratory data mining",2010

[36]- MFCC, http://en.wikipedia.org/wiki/Mel-frequency_cepstrum

[37]- Serkan Kiranyaz, Ahmad Farooq Qureshi and Mocef Cabbouj, Senior Member, IEEE, "A Generic Audio Classification and Segmentation Approach for Multimedia Indexing and Retrieval", May 2006.

[38]- L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition" IEE, vol 77, pp 257-286 (1989)

[39]-H. Demuth and M.Beale, "Neural Networ ToolBox User's Guide"(MathWorks 1998)

[40]- T. Cover and P.Hard, "Nearest Neightbor Pattern Classificaction", IEE (1967)

[41]- http://es.wikipedia.org/wiki/Sonido#La_voz_humana